Novel Discretization Schemes for the Numerical Simulation of Membrane Dynamics

DISSERTATION

Kyle F. Kolsti, Lieutenant Colonel, USAF

AFIT/DS/ENY/12-03

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this document are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT/DS/ENY/12-03

NOVEL DISCRETIZATION SCHEMES FOR THE NUMERICAL SIMULATION OF MEMBRANE DYNAMICS

DISSERTATION

Presented to the Faculty
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy

Kyle F. Kolsti, BSCE, MSCE
Lieutenant Colonel, USAF

September 2012

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

AFIT/DS/ENY/12-03

NOVEL DISCRETIZATION SCHEMES FOR THE NUMERICAL SIMULATION OF MEMBRANE DYNAMICS

Kyle F. Kolsti, BSCE, MSCE
Lieutenant Colonel, USAF

Approved:

Donald L. Kunz — 23 Aug 2012
Donald L. Kunz, PhD (Chairman) — Date

23 Aug 2012
Eric D. Swenson, PhD (Member) — Date

23 Aug 2012
Matthew C. Fickus, PhD (Member) — Date

Accepted:

MU Thomas — 3 Sep 2012
M. U. THOMAS — Date
Dean, Graduate School of Engineering
and Management

AFIT/DS/ENY/12-03

Abstract

Motivated by the demands of simulating flapping wings of Micro Air Vehicles, novel numerical methods were developed and evaluated for the dynamic simulation of membranes.

For linear membranes, a mixed-form time-continuous Galerkin method was employed using trilinear space-time elements. Rather than time-marching, the entire space-time domain was discretized and solved simultaneously. Second-order rates of convergence in both space and time were observed in numerical studies. Slight high-frequency noise was filtered during post-processing.

For geometrically nonlinear membranes, the model incorporated two new schemes that were independently developed and evaluated. Time marching was performed using quintic Hermite polynomials uniquely determined by end-point jerk constraints. The single-step, implicit scheme was significantly more accurate than the most common Newmark schemes. For a simple harmonic oscillator, the scheme was found to be symplectic, frequency-preserving, and conditionally stable. Time step size was limited by accuracy requirements rather than stability. The spatial discretization scheme employed a staggered grid, grouping of nonlinear terms, and polygon shape functions in a strong-form point collocation formulation. The observed rate of convergence was two for both displacement and strain. Validation against existing experimental data showed the method to be accurate until hyperelastic effects dominate.

AFIT/DS/ENY/12-03

To my amazing wife, Colleen, who is still the smartest person in the house, for her loving encouragement and support, and to my beautiful children, Matthew and Kathleen, whose innocence and constant search for fun kept everything in perspective.

Acknowledgements

I am sincerely grateful to my research advisor, Dr. Donald Kunz, for patiently guiding me through this research, and for giving me the freedom to pursue ideas while simultaneously keeping me from wandering too far off track.

I also wish to thank Dr. Bret Stanford from the Air Force Research Laboratories, for his valuable feedback and advice.

There are many former commanders, mentors, and friends who went out of their way to help me have this incredible opportunity. I hope to either repay your kindness or pay it forward.

Finally, I would like to acknowledge the Air Force Office of Scientific Research for funding this research.

Kyle F. Kolsti

Table of Contents

Page

Page

List of Figures

Figure Page

List of Tables

List of Symbols

Symbol		Page
ν	Poisson ratio	4
w	Transverse displacement (also labeled u_3 in some places)	9
$\bar{\rho}$	Density (mass per unit area)	10
P	Tension	10
c	Solution propagation speed	11
L	Lagrangian	13
T	Kinetic energy	13
U	Potential energy	13
q	Generalized coordinates	13
p	Generalized momenta	13
H	Hamiltonian	13
$\mathbf{u}$	Displacement	21
$\mathbf{M}$	Mass matrix	21
$\mathbf{C}$	Damping matrix	21
$\mathbf{K}$	Stiffness matrix	21
β	Newmark method parameter	22
γ	Newmark method parameter	22
T	Period	27
Ω	Spatial domain	28
σ	Generalized stress	29
f	External pressure force	31
ξ	Local coordinate	35

NOVEL DISCRETIZATION SCHEMES FOR THE NUMERICAL SIMULATION
OF MEMBRANE DYNAMICS

I. Introduction

The advent of the Micro Air Vehicle (MAV) has forced the scientific community to focus on a familiar but extremely complex mechanism: the flapping wing. Well suited for low-speed or even hovering flight, the flapping wing provides thrust, lift, and control. It is typically highly flexible and non-uniform materially and geometrically. Besides undergoing large and rapid rigid-body motion, it also experiences large deformations. The true aeroelastic system is therefore highly nonlinear with coupled structural and aerodynamic phenomena [27, 45]. The aerodynamics alone have proved more difficult to model than those of a fixed wing [93].

Micro Air Vehicles often utilize flapping wings with membranes as the lifting surfaces. Structurally, a common flapping wing design consists of a main spar, stiffeners, and a thin flexible membrane to serve as the lifting surface. Fabricated MAVs tend to employ much simpler variations on the theme. The spar and stiffeners (often called battens) have bending and torsional stiffness and can be considered to behave as beam elements, leading to the current nomenclature "beam-membrane structure." They preserve the form and provide the forcing mechanism, while the membrane provides the lightweight lifting surface.

As interest in flapping membrane-based wings intensifies, requirements for numerical simulation will grow. Lessons learned from two-dimensional airfoil studies will naturally progress to fully three-dimensional subjects for validation and application. As the computational expense grows accordingly, demand will also rise for novel and efficient discretization schemes that retain or even improve sufficient solution accuracy. This forecasted need motivated the present research, drove its objectives, and precisely defined its scope.

Background and Motivation

Compared to a rigid airfoil, a flexible membrane's passive reactions to aerodynamic loads can improve gust response [87] and stall characteristics [119]. Wind tunnel experiments of a membrane stretched over a fixed frame have revealed the significance of at least five vibration modes, with strong interactions between the membrane oscillations and vortex shedding [113]. Gordnier [45] performed high-fidelity numerical simulations of a similar configuration and observed both standing and travelling wave responses in the membrane depending upon the flow regime, leading him to conclude that an advanced multidisciplinary approach was necessary to fully grasp the complicated system.

Both air loads and inertial forces contribute to structural deformation. To examine their relative significance, Yin and Luo [143] defined the mass ratio m^*, which in physical terms is the ratio of inertial force of the wing and the aerodynamic pressure. They referred to an earlier study of the hawkmoth [27], where $m^* = 5$ and deflections were similar in a vacuum and in air, indicating that inertial forces dominate. In contrast, dragonfly wings are much lighter at $m^* = 1$ and aerodynamic forces dominate. Another study of the hawkmoth [121] showed a 23% difference in modal frequencies between air and vacuum experiments, leading them to conclude some aerodynamic coupling was present. Hawkmoth wings have been found to deform primarily due to inertial forces, and damping could approximate the aerodynamic effects [27]. Clearly the structural dynamics are a critical part of this complex system, and must be suitably modeled to achieve a useful outcome.

In nature, many insects [139] and bats [128] have beam-membrane wings. Because they have mastered flight through evolutionary development, these living MAVs have been the subject of many studies. Attempts to capitalize on their success despite our limited knowledge have also led to many "bio-inspired" designs. One of the most common subjects is the hawkmoth (*Manduca sexta*). Its relatively large size is helpful because it enables detailed experimental studies [138], and evokes a suitable size for fabricated, missionized vehicles with payloads. Its flapping mechanism – forcing only at the shoulder with passive control through venation variations – avoids the complex control mechanisms of actively-deformed wings like those in birds and bats. Conventional structural analysis can therefore be performed on the wings. The prevalence in the literature of the hawkmoth and the models it inspired motivates this study to focus on membrane systems employed by this class of MAVs.

Research Goals

As the demand for higher-fidelity aeroelastic simulations of membrane-based wings continues to grow, so will the need for practical structural models that are capable of capturing the vitally important membrane dynamics. In anticipation of this need, the purpose of this study was to devise and evaluate numerical schemes to accurately simulate the dynamics of a membrane like those employed in typical Micro Air Vehicle wings. Only the membrane component was addressed in this study because, as mentioned previously, the complexity of an entire flapping wing system requires an interdisciplinary approach. The numerical schemes developed here will be demonstrated using fixed or prescribed frames only. Having thoroughly verified and validated the results, future incorporation of the schemes into flapping beam-membrane structures is a natural progression.

The scope and requirements for this study will now be delineated. The hyperbolic governing partial differential equations (PDEs) for a membrane are second order in space and time (*i.e.*, wave equations). All proposed schemes were to be capable of simulating the transient response of a membrane, including standing and travelling waves. Modal analysis was therefore expressly excluded – if needed, modal frequencies could be extracted from the time series information of the transient response. Accuracy was to meet or exceed that of commonly-used numerical computation methods.

Capability requirements and scope of suitability for the new numerical schemes were derived from the characteristics of the physical system under consideration: a fabricated (as opposed to biological) structure consisting of a thin, isotropic membrane stretched across a rigid frame. Selecting this system enabled the design to focus on a narrow but useful design space. The model was therefore customized for systems abiding by the following assumptions:

- Geometry
    - Large deformation/small strain (at least geometrically nonlinear)
    - Variable geometry (not restricted to circle, square, *etc.*)
    - Membranes must be planar when at rest
- Membrane properties
    - Isotropic
    - Buckling and wrinkling were neglected
    - Linear elastic material model
    - Poisson ratio ν: 0 to 0.5 (hawkmoth approximation $\nu = 0.49$ [27])
    - Membrane at rest may be slack or prestressed
    - Clamped at all boundaries (Dirichlet boundary conditions only)

- Forcing functions
  - All external and body forces are distributed and smooth (discontinuous point forces not expected)
  - Directional forces (tractions or body forces like gravity)
  - Follower forces
  - Tightly-coupled aerodynamic forces that respond to the membrane shape
  - Time-dependent forces
- Frame
  - Prescribed motion only. Incorporation of a coupled beam model for the rigid frames was beyond the scope of this study.

The performance of the proposed techniques was assessed through a rigorous progression of verification and validation. As precisely defined in Ref. [112], the verification process ensures that a model is solving the chosen equations correctly and consistently. After code has been verified, validation addresses whether the proper equations were selected for approximating the physical system under consideration. Experimental data therefore plays a key role in validation.

A wide variety of methods for building a simulation that meets the listed requirements are available, and many of them will be discussed in the literature review in Chapter Two. For two-dimensional membranes of arbitrary geometry, the Galerkin finite element method (FEM) is the predominant approach. The most commonly associated time integration scheme is the Newmark family of methods, favored mostly for the unconditional stability of its implicit average acceleration scheme. Because of the Galerkin formulation's numerical integration procedures, the computational expense of this approach for nonlinear dynamic problems is significant. In addition, for highly nonlinear problems, the stability benefits of the implicit Newmark methods are degraded.

The narrowed design space for this study, however, provided the opportunity to develop new methods tailored to the requirements. Hence, novel schemes for both time and space discretization were created, evaluated, and incorporated into a final membrane model. In particular, a time-marching scheme based on Hermite polynomial interpolation was developed that is significantly more accurate than the second-order Newmark methods, yet it is stable at time step sizes appropriate for capturing the relevant dynamics. Likewise, a spatial discretization scheme based on the point collocation method was devised that handles the geometric nonlinear membrane behavior while avoiding the requirement for numerical integration. When put together, these two schemes provided a practical, robust model for the dynamic simulation of geometric membranes.

Organization

Thus far the motivating systems and the scope of the study have been outlined. The remainder of the dissertation will detail the individual components of the model and finally evaluate the complete membrane model. The literature review will be conducted in Chapter Two to frame the new methods in the context of conventional approaches. Chapter Three details the development and evaluation of the Simultaneous Time-Continuous Galerkin (STCG) method for a linear membrane. Chapter Four progresses to fully nonlinear membranes. The jerk-based constraint formulation for Hermite time interpolation is proposed and evaluated, followed by a similar development of the staggered-grid point collocation model. Having rigorously examined these two schemes individually, they are finally incorporated into the final membrane model for analysis. Lastly, in Chapter Five, results will be summarized and discussed from the perspective of the study objectives. Recommendations for further study will also be offered.

II. Literature Review

What exactly is a membrane? Jenkins provides excellent definitions depending on one's perspective: a *membrane* is a structure that does not resist bending, while a *membrane model* is "an idealized model of a plate or a shell structure, wherein the in-plane response dominates away from domain and load boundaries." [63]. With its total lack of bending stiffness, a membrane may be thought of as a two-dimensional version of a string or cable [95]. When subjected to an external load, the only restoring force is from the tension in the plane of the membrane. Membranes are inherently nonlinear structures that may undergo large rigid body displacements and rotations, and often large strains as well. Thorough discussions of membrane mechanics and challenges may be found in [64], [63], [96], and [95].

The prototypical problem for static membrane study is the circular membrane with fixed circumference that is inflated under constant pressure. Commonly called the Hencky problem [51], this configuration facilitates experimentation and analytical solutions. Finite element (FE) models have been applied to the Hencky problem that use detailed geometric surface descriptions with Jaumann strains and stresses [96]. Another approach is to use simple linear-elastic elements applicable to the large displacement/small strain regime [97].

Despite the intrinsic nonlinearity of true membranes, simplifying assumptions may be appropriate for some applications. Based on these possible assumptions, a membrane model can be classified as "linear," "small strain/finite rotation," or "fully nonlinear" based on how (or whether) it handles the geometric and material nonlinearities [64]. Stanford *et al.* [125] referred to these three categories as "low-", "medium-", and "high-fidelity."

- *Linear* (*Low fidelity*). This model is the simplest, but carries the most stringent assumptions. The material is linear elastic. Initial membrane internal forces (pre-tension) are significantly greater than any change in internal forces caused by deformations. External loads are not affected by the membrane's shape. Only transverse displacements, w, are permitted, and the displacements are small. The linear model may be appropriate only in cases with small pressures, small displacements, and large initial tensions [126, 125]. The linear model is invalid (unbounded) for a slack membrane [125]. Further, it has been found that a circular membrane that is planar when undeformed cannot be linearized, no matter how small the displacements [88].

- *Small strain/finite rotation* (*Medium fidelity*). Also called a "geometrically nonlinear" model. In this case, the membrane's resistance to an increase in transverse load (a force normal to the plane of the membrane) depends on the current shape of the membrane. For example, if the membrane is flat, the membrane offers no resistance when a load is applied. On the other hand, if the membrane is deformed out-of-plane, the surface is sloped such that tension forces are able to resist further deformation when the load is incremented. This phenomenon occurs even though the material is considered to be linear elastic.

- *Fully nonlinear*(*High fidelity*). In a fully nonlinear model, both material and geometric nonlinearities are accommodated. Often, the material is considered to be hyperelastic – the strain energy depends on both the initial and the deformed state, but is not dependent on the path between those states [15].

Geometrically nonlinear models have been shown to be suitable until hyperelastic effects begin to dominate. For an initially flat, circular, rubber membrane with a clamped circumference, this transition occurs at a center deflection of approximately

25% of the membrane's radius [106]. Since this degree of deformation was considered unlikely for typical Micro Air Vehicle wings, the fully nonlinear model was not considered in this study.

With the fundamental characteristics of membranes in mind, the rest of this chapter reviews numerous approaches for modeling a membrane. Broadly speaking, they can be categorized as analytical or numerical. Methods for handling the temporal dimension in a numerical scheme are also discussed.

Analytical Models

Analytical solutions for membranes are restricted to special subcases where forcing functions, initial conditions, and boundary conditions are tractable. As a result, configurations are typically rectangular for Cartesian coordinates, or circular for polar coordinates or to capitalize on radial symmetry.

The linear membrane model is most amenable to analytical treatment, and is therefore most prevalent in the historical and canonical literature [91, 136, 46, 108, 133]. The governing equations may be obtained by either a variational approach using energy formulations and Hamilton's Principle, or a Newtonian force equilibrium approach. The membrane's surface is defined by the displacement $w(x, y, t)$ of a particle (x, y) in the z direction. Letting $\bar{\rho}$ be the membrane's constant density (mass per unit area), P be the constant tension (force per unit length), and $f(x, y, t)$ be the external pressure in the same direction as w (force per unit area), the well-known second-order PDE is

$$\bar{\rho}\frac{\partial^2 w}{\partial t^2} = P\nabla^2 w + f \tag{1}$$

with the Laplacian operator defined as

$$\nabla^2 \equiv \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \tag{2}$$

The wave speed c, where $c = \sqrt{P/\bar{\rho}}$, is variously called the the characteristic velocity, the speed of sound for the material [28], or the solution propagation speed [58]. For the static case, the Poisson equation is recovered. If shear forces are included, the internal forces are represented by N_x, N_y, and N_{xy}, and the static version of Eq. 1 may be presented as [125]

$$N_x \frac{\partial^2 w}{\partial x^2} + N_{xy} \frac{\partial^2 w}{\partial x \partial y} + N_y \frac{\partial^2 w}{\partial y^2} + f = 0 \tag{3}$$

Exploiting the radial symmetry of circular membranes permits a two-dimensional development to solve a three-dimensional problem. The complementary energy principle has been used to derive a fully nonlinear model for predicting large deformations [88]. Likewise, equilibrium principles have been employed with assumed functions for the dependent variables; the solution is obtained after solving for the constant coefficients multiplying the assumed functions [35].

The perturbation method has also been effective for membrane analysis. A perturbation method of the pressure term has been used to analyze a small strain/finite rotation circular membrane. The numerical solution was obtained through a Taylor expansion of the analytical solutions to differential equations [127]. A perturbation of the external force to the same problem has also yielded accurate results [32]. For the dynamic case, the vibrations of a large-displacement nonlinear square membrane were accurately modeled by using the perturbation method to derive simplified approximate governing equations [25].

Numerical Models

The simplifying assumptions required by analytical approaches are quickly violated when more complex systems are considered. The inclusion of arbitrary domain geometry or interactive external forces, for example, require a numerical approach to approximate the solution. The three primary approaches are finite differences (FD), finite volume (FV), and finite element analysis (FEA) [74]. Although all three approaches can be universally applied to approximate the solutions to differential equations, there are strengths and weaknesses associated with each. As a result, the first two approaches are generally employed by the fluid dynamics and heat transfer communities, and FEA is heavily favored by structural analysts. For this reason, this dissertation will focus primarily on FEA. However, certain applications of one method can lead to similarities (if not overlap) with another, so we should resist the tendency to pigeonhole methods to one application or another.

The computational expense of a particular model is affected largely by the phenomena the analyst wishes to include. Thus, one should strive to use the most efficient path that produces answers of sufficient accuracy. In one extreme, rigid plates have been used for kinematic optimization, where the minimizing computational effort was paramount [13]. This is the exceptional case, however – estimates of aerodynamic power, work, thrust are commonly sought and require a non-rigid body to capture the physics. The numerical simulation of a flexible, dynamic, nonlinear structure is a challenging endeavour, and following sections will review available methods.

Classical Mechanics and Hamiltonian Systems.

Classical mechanics form the mathematical basis for the variational development of finite element models and spring-mass models. A brief review will be presented here as a foundation for later developments. As opposed to the force-oriented Newtonian mechanics, classical mechanics apply variational calculus to scalar energy functionals to determine the optimal solution of a system. The classical Lagrangian formulation is well known [42, 78, 44], and leads to Hamilton's canonical equations. More recently, direct solutions have been obtained using Hamilton's Weak Principle, which is derived from Hamilton's Law of Varying Action (HLVA) [80]. Both of these approaches have been utilized to derive numerical methods for approximating the behavior of Hamiltonian systems.

The Lagrangian L of a mechanical system is defined as the difference between the system's kinetic energy T and potential energy U. Let q be the vector of generalized coordinates according to the Cartesian position X and time t. Then

$$L(t, q, \dot{q}) = T(\dot{q}) - U(q) \tag{4}$$

The physical path of the system from time t_0 to time t_1 satisfies the Extended Hamilton's Principle [118, 108],

$$\delta \int_{t_0}^{t_1} L dt + \int_{t_0}^{t_1} \overline{\delta \mathcal{W}} dt = 0 \tag{5}$$

To transform the formulation from Lagrangian to Hamiltonian, we next define the generalized momenta p of the generalized coordinates:

$$p_i = \frac{\partial L}{\partial q_i} \tag{6}$$

The Hamiltonian H is then defined as

$$H = \sum_{j=1}^{n} \dot{q}_j p_j - L \tag{7}$$

and with our formulation of T it can be proved that $H(q, p) = T(p) + U(q)$. Substitution into the functional I gives the equation in terms of the Hamiltonian:

$$I = \int_{t_0}^{t_1} \left(\sum_{j=1}^{n} \dot{q}_j p_j - H \right) dt \tag{8}$$

Finding the extremum of this functional, one obtains the Hamilton equations of motion (also called the“canonical equations of Hamilton” [78]).

$$\dot{q}_i = \frac{H(q,p)}{p_i}, \quad \dot{p}_i = -\frac{H(q,p)}{q_i} \tag{9}$$

For direct numerical application, the configuration state vector η is defined to contain the coordinates q and momenta p,

$$\eta = \left\{ \begin{array}{c} \{q\} \\ \{p\} \end{array} \right\} \tag{10}$$

leaving H as a function only of η. The partial derivatives of H with respect to η are

$$\frac{\partial H(\eta)}{\partial \eta} = \left\{ \begin{array}{c} \partial H/\partial q_1 \\ \dots \\ \partial H/\partial q_N \\ \partial H/\partial p_1 \\ \dots \\ \partial H/\partial p_N \end{array} \right\} \tag{11}$$

The canonical equations can then be put into matrix form, which is more amenable to numerical implementation.

$$\dot{\eta} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{I} & \mathbf{0} \end{bmatrix} \frac{\partial H(\eta)}{\partial \eta} \tag{12}$$

The block matrix is often labeled **J** (not to be confused with a Jacobian matrix), resulting in the form [44]

$$\dot{\eta} = \mathbf{J}\frac{\partial H(\eta)}{\partial \eta} \tag{13}$$

With simple difference equations, this form can easily be turned into a midpoint value integration scheme [22, 80],

$$\frac{\eta_1 - \eta_0}{\Delta t} = \mathbf{J}\frac{\partial H(\frac{\eta_0+\eta_1}{2})}{\partial \eta} \tag{14}$$

The second approach, direct solution based on Hamilton's Weak Principle (HWP), retains boundary terms in Hamilton's Law of Varying Action that are neglected in Hamilton's Principle [7]. By honoring these terms, the energy balance through a time step is preserved. The weak form captures all of the boundary conditions in a single functional and reduces the interpolation order requirements. Constant shape functions may be permissible, leading naturally to discontinuous Galerkin formulations where the integration can be performed by inspection [4, 80]. In fact, weak Galerkin formulations and HWP result in equivalent schemes [18]. Representing non-conservative generalized forces as Q, Hamilton's Weak Principle as developed in [80] is

$$\int_{t_0}^{t_1}\left(\sum_{j=1}^{n} p_j\delta\dot{q}_j - \sum_{j=1}^{n} q_j\delta\dot{p}_j - \delta H + \sum_{j=1}^{n} Q_j\delta q_j\right)dt - \sum_{j=1}^{n} p_j\delta q_j|_{t_0}^{t_1} + \sum_{j=1}^{n} q_j\delta p_j|_{t_0}^{t_1} = 0 \tag{15}$$

Statements of HWP may also be found in [18, 16, 56, 57].

Spring-Mass Models.

Spring-mass models of a membrane discretize a continuum into a system of interconnected particles. Movement of the particles determines the kinetic energy of the system. Mass lumping at the particles is intrinsic (the consistent mass treatment of FEM is not an option) and the rotational momentum about points between the particles is ignored. However, for most problems and with small elements, the rotational inertia is insignificant relative to the translational inertia [134]. Massless springs connect particles and house the potential energy. The model intrinsically handles geometric nonlinearities and (through the spring model) material nonlinearity. Systems can range from a simple harmonic oscillator to a cloth blanket draped over a table. Spring mesh models offer the advantages of simplicity and computational speed [132]. These factors make them popular in the field of animation, where they have found the most use and development [73]. One must be aware, though, that in this milieu realistic-looking results may be the objective rather than physically accurate ones [72]. Despite this caution, the field offers interesting alternatives to traditional techniques in solid mechanics.

One study [73] deemed spring-mass models to be superior to traditional FE formulations for purely axial structures undergoing large displacements, in particular the hanging chain or net. Algorithm details were not presented other than stating that Runge-Kutta time integration was used. Demonstrations of three-dimensional static problems showed the technique to be promising but with some limitations (in particular, computational expense and potential instability in complex problems).

According to Gelder, as of 1998 little work had been done to compare results of spring mesh models with those of traditional finite element models [132]. He evaluated the capability of a spring mesh model to model an isotropic linearly elastic membrane undergoing in-plane deformations. All simulations were static analyses. He compared the stiffness matrices of the spring mesh model and the constant strain model to prove the spring mesh cannot exactly match the constant strain model when the springs have the same stiffness. In particular, the spring model does not include the Poisson effects. Based on the geometry of an element and the desired edge stresses, he modified the spring stiffness coefficients to more closely match the constant strain triangle model. Exact solution was still not possible.

Delingette [31] rigorously developed objects called “triangular biquadratic springs” in such a way that the membrane strain energy was exactly that of an equivalent continuum model. On an unstructured triangular mesh, these springs effectively modeled static non-linear membrane deformations. As contrasted with previous efforts, this study included an angular stiffness to the springs. This mechanism enabled inclusion of the Poisson effect.

Finite Element Models.

The predominant numerical method for membrane simulation is the finite element method (FEM). For MAV applications, different approaches have included distinct beam and shell elements [90, 34], membrane elements with varying parameters [86], or plate elements with bending stiffness set to near zero for membrane portions of the structure [124].

In FEA, membranes are commonly considered subcases of shells [95] and element development follows that train of thought. However, the nature of a membrane structure (light weight, minimal thickness, large displacements and rotations, *etc.*) causes unique problems for conventional FEA. While most production codes have membrane elements incorporated, their accuracy may be questionable for some cases; in particular, complex phenomena such as wrinkling may not be accurately predicted [95]. As a result, attempts have been made to custom tailor elements to this problem.

Pauletti [97] refined a membrane element designed specifically for geometrically nonlinear membranes. The original element is called “Argyris’s Natural Membrane Finite Element” and is based on the constant strain triangle. The model splits the element stiffness matrix into three components that capture geometric nonlinearities, constitutive relationships, and external load effects separately. The membrane is assumed to be linearly elastic. Comparisons to benchmark cases demonstrated good performance. The linear membrane displayed limited accuracy under large deformations. Leung [83] developed trapezoidal finite elements using trigonometric shape functions instead of polynomials. Modal frequencies were obtained more accurately than with the conventional polynomial approach, and with greater conditioning and stability.

Clearly there is no consensus about how to tackle membranes in FEA, and special care must be taken to capture the unique behavior of a particular membrane problem.

Point Collocation and Group Finite Element Methods.

The model developed here will utilize two approaches which are relatively rare in structural FEM applications: the point collocation method and the group finite element (FE) formulation. Both approaches were selected for this effort because they tend to result in simpler formulations [26, 146], potentially offering greater code flexibility without sacrificing accuracy. Coincidentally, both the point collocation method and group FE formulation have close ties with the development of computational fluid dynamics (CFD) schemes. Development of the group FE formulation in the literature centers around fluid mechanics applications and examples [26, 37, 38], and the similar lumping of nonlinear terms in the flux vector is standard practice for finite volume formulations [55]. Similarly, the point collocation method shares a history with and bears a resemblance to the finite difference method [146]. The nonlinear wave equations that describe membrane dynamics offer an interesting test case for further examining how well these CFD-associated techniques transfer into the structural dynamics milieu.

The collocation method is commonly mentioned in the finite element literature when listing the members of the Method of Weighted Residuals family. However, it is rarely seen in practice. The weighting function for each designated point in the domain is the Dirac delta function. By definition, the Dirac delta function equals zero everywhere except at its associated point, and its integral over the domain equals one [146]. The resulting system of equations solve the PDE (Partial Differential Equation) point-wise rather than in an integral sense. Thus, there is no need for Gaussian integration over an element. Posed in the strong form, the method requires the interpolation scheme to be differentiable to the same order as the PDE [146, 1], but the formulation of the system of equations has been found to be less complicated than when employing the Galerkin method [1].

Also, while certainly applicable to a conventional FEM mesh, the point collocation method lends itself to a variety of unconventional discretization and interpolation schemes. Element shapes can expand beyond triangles and quadrilaterals to n-sided polygons. Taking the concept even further, "meshless" [10, 61] and "element-free" [11, 76] methods use least-squares fits, radial basis functions, or other neighboring-node-based techniques for forming the system of equations. Using a meshless point collocation method, [1] solves a wide variety of problems including heat conduction, Couette flow, and a cantilever beam. In the interest of computational efficiency for the dynamic simulation, the present model utilizes a staggered background mesh so it does not fall into the meshless category.

In a group finite element formulation (also called "product approximation" [26]), aggregated nonlinear terms are first computed, then interpolated as a single degree of freedom. Consider the term ρuv where ρ, u, and v are each dependent variables [38]. Rather than applying trial solutions ϕ_j to each of the variables, the group formulation interpolates their precomputed products as $\sum \phi_j \left(\rho uv\right)_j$. Significant computational savings have been observed, with the benefits increasing from higher dimensionality or order of nonlinearity [37]. For some cases, point-wise accuracy may actually be higher than that of the Bubnov-Galerkin method [26]. This observation hints at a potential synergy in the pairing of the group FE formulation with the point collocation method.

Time Integration Methods

Numerical approximation of initial value problems (IVP), also aptly called evolutionary differential equations [3], has been studied for centuries, with Euler's foundational work coming in the 18th Century. Despite its rich history, the challenge of balancing stability and accuracy continues to invite further consideration, especially as computational models are applied to progressively more complex systems.

A primary source of problems for evolutionary equations is the propagation and build-up of errors. Hence, dissipation of unwanted high-frequency content is desired. Ideally it is achieved through the formulation of the predictive algorithm, often employing parameters to tailor the behavior [54]. In some circumstances, this dissipative behavior may be necessary because noise can significantly affect extrapolated predictions and result in instability [58]. Spurious high-frequency oscillations have long been documented for Galerkin solutions of the wave equation [92] and the shallow water wave equations [5, 39]. The time-discontinuous Galerkin (TDG) method offers additional flexibility through the flux resolution to control dissipation [53]; upwinding techniques for the jump operators in particular have proven effective in improving stability [59, 16].

Structural dynamics formulations spring from the equilibrium equation

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} + \mathbf{F} = 0 \tag{16}$$

where $\mathbf{u}$ is the solution vector, $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$ are the mass, damping, and stiffness matrices respectively, and $\mathbf{F}$ is the external force vector.

The most common time integration technique in FEM is direct integration. The direct integration approach splits the domain into two parts: a spatial component to be addressed with a finite element approximation and a time-marching predictor algorithm. The FEA is a static solution where for linear systems the forces exerted by the structure are a function of the structure's current configuration. Given appropriate initial conditions, external forces, and the internal forces from the FEA, the time-marching algorithm predicts the configuration at the end of a time increment. That solution then provides the initial conditions for the next time step, and so on. Much effort has gone into developing algorithms that balance the often contradictory demands placed upon them. Numerous families of algorithms have been developed, including the single-step direct integration methods that are either explicit (such as the Euler and central difference methods) or implicit (Newmark, Wilson, *etc.*). These procedures produce at best second-order accuracy [58]. Explicit methods are more economical at each step, but they are conditionally stable and typically require more steps. Implicit methods require more expensive calculations but can be formulated to be unconditionally stable, enabling larger time steps.

Besides stability limits, the time step size for nonlinear systems is limited by other factors. A chaotic system may force strict accuracy requirements to minimize the propagation of numerical perturbations. For implicit schemes, iterations may not converge if the step size is so large that a sufficiently good initial guess cannot be made. These requirements may limit step size to the point that conditionally stable schemes become feasible. From these considerations arose the goal of implementing a method that offers higher accuracy than the predominant Newmark methods. Effective handling of nonlinearities, conservation of energy, improved frequency preservation, and useful error estimation for step size or damping control were additional objectives.

In the context of structural finite element analysis, the conventional approach for marching this type of model through time is to use one of the members of the well-documented Newmark family [58, 68, 28]. All of the Newmark methods are single-step methods that are distinguished by the values chosen for two parameters, β and γ. The two most commonly used methods are the explicit, conditionally stable central difference method and the implicit, unconditionally stable trapezoidal rule (also called the average acceleration method). Both methods are second-order accurate in time. Unconditional stability is generally attractive for structural applications—despite the presence of high-frequency content or numerical noise, large time steps can be taken while preserving the accuracy of the relevant low-frequency modes. It should be noted that unconditional stability is no longer guaranteed for non-linear problems and the order of the methods may drop to one in the presence of damping [28]. Despite the popularity of the Newmark methods in structural analysis, other time integration techniques such as the Runge-Kutta method and space-time finite element formulations have been successfully applied, showing there is room for alternatives.

The Runge-Kutta technique is a multi-step method that is most commonly applied in Computational Fluid Dynamics or Multi-Body Dynamics. As an explicit procedure, it is computationally efficient and the myriad of schemes provide many alternatives in the tradeoff between accuracy and speed. Runge-Kutta methods are well-known and covered extensively from numerous viewpoints, including differential equations [48, 142], numerical methods [71, 77, 3, 103, 117], and computational fluid dynamics (CFD) [14, 55]. Examples of its application with conventional FEA are few [94, 107]. However, the Runge-Kutta method is commonly employed along with the Discontinuous Galerkin Method [85, 53]. The Runge-Kutta method discretizes the first-order differential equation in time to generate explicit functions for slope in the form $\Delta U/\Delta t = H(U)$. The slope is calculated at optimized points within the time step. Weighting coefficients refine the final prediction of U at the end of the time step.

Symplectic Methods.

The symplectic methods, also called geometric or variational integrators, retain the relationship between the generalized coordinates q and their generalized velocities or momenta p. Thus the flow or mapping of the system from an initial point $(q,p)_0$ to its future configuration $(q,p)_1$ should preserve the area of a given set of points as they move from $t = t_0$ to $t = t_1$ [48]. In a discretized Hamiltonian system the coordinates and momenta are related to the Hamiltonian as $\Delta p/\Delta t = -\Delta H(q,p,t)/\Delta q$ and $\Delta q/\Delta t = \Delta H(q,p,t)/\Delta p$. Often in mechanics the potential energy is a function only of position ($U = U(q)$) and the kinetic energy is a function only of velocity ($T = T(p)$). A variational integrator iterates between q and p across multiple stages to achieve a prediction for the future configuration vectors q and p. Because the relationship between q and p is largely preserved, symplectic integration has been shown to more effectively conserve a system's energy over time. This feature is of utmost importance during long-duration simulations where small rates of energy gain or loss result in substantial errors at the end.

For a thorough treatment of symplectic integration, refer to [48, 137, 84]. Also, a seminal work in the field of symplectic integration is that of Yoshida [144]. Yoshida developed a rigorous method for deriving exact coefficients for high order integrators.

Symplectic integration has been successfully integrated into FE models of beam vibration. Leung and Mao [82] demonstrated the use of a symplectic algorithm in the FEA of non-linear vibrations of a beam and recommended it for use in plates and shells. They separated the kinetic and potential energy equations to utilize the midpoint symplectic technique [22], which they called the "time-centered Euler scheme." Dash *et al.* [30] used Yoshida's 8th order scheme and observed good results but marked instability above a critical Δt. Note that some of the other methods, such as the Newmark and certain Runge-Kutta formulations, can be shown to be symplectic [68, 84, 137]. Hughes mathematically proved that for an unforced undamped response the Newmark trapezoidal rule exactly conserves the total energy of a system [58].

The fact that a method is symplectic does not necessarily guarantee increased performance or suitability for a given problem. Energy preservation may indeed be counter to a need for algorithmic dissipation. Thus, in this research, symplecity will be examined to gain further understanding of proposed methods, but will not be a basis for their formulation or an objective to be pursued.

Space-Time Finite Elements.

Space-time finite element formulations have been seen as a viable alternative to direct integration for several decades [9, 17, 100]. In this method, the elements span not only space but also a discrete time increment. The most frequently encountered space-time finite element technique is the time-discontinuous Galerkin (frequently abbreviated as TDG [62]) method. In the discontinuous Galerkin method, element shape functions produce different values of the field variable at their interfaces; flux schemes are then employed to resolve the ambiguity. High-order shape functions using a variety of basis types are permitted. For example, in a wave problem, basis functions that are themselves solutions to the wave equation have provided great accuracy while minimizing computational cost [101]. In a different but equally effective approach, Hamilton's Weak Principle has been applied to develop simple elements with constant and linear shape functions. The integration could be performed by inspection, and led to a system of algebraic equations for solving a variety of nonlinear dynamic problems [4, 56].

Implementation of space-time FEA in all of the aforementioned studies involved obtaining the solution for one slab of elements at a time, where a slab is defined as the set of elements in a given time increment t_0 to $t_1 = t_0 + \Delta t$. The nodal values at t_1 were then used as initial conditions for the following time increment. In contrast, the method employed during this research for modeling linear membranes solved the entire space-time domain simultaneously.

Hermite Time Interpolation.

An alternative time integration method was developed for this study: Hermite polynomial interpolation. A unique form of collocation, this scheme uses derivatives at the end points to provide the necessary number of equations rather than additional interior points [81]. A common example of Hermite polynomials in finite element analysis can be found in simple beam elements, where displacements and rotations at the ends of the beam define the cubic polynomial shape functions [107, 28]. Higher-order Hermite interpolations are possible, but they demand additional end point function derivatives, interior nodes, or some other means of providing enough constraints to define a unique solution.

For initial value problems in general, a wide variety of time-marching methods are defined by the selection of a polynomial interpolant and a corresponding numerical integration scheme. These include linear multi-step methods and single-step collocation methods like the implicit Runge-Kutta methods [48, 3]. The ubiquitous implicit midpoint rule also falls under this category [116]. These methods rely on internal collocation points for their definition. In contrast, Hermite polynomials can be cast in a two-point form, so the interpolation is defined by the function value and its derivatives at the end points only. The term "prolongation" has been used to refer to the use of additional derivatives at the end points to provide the necessary number of equations rather than additional interior points [81]. A common example of Hermite polynomials in finite element analysis can be found in simple beam elements, where cubic shape functions are uniquely defined by the displacements and rotations at the ends of the beam [107, 28]. Higher-order interpolations are possible, but they demand additional end point function derivatives. One can also obtain a solution by replacing one of the derivatives by a time integral constraint, called an eliminant [77].

Previous results using Hermite time interpolation have shown excellent accuracy relative to the central difference and trapezoid methods. Leok *et al.* [81] used quintic interpolation polynomials in their prolongation-collocation method. The variational integrators were derived by applying Euler-Maclaurin quadrature to the Lagrangian formulation. It was mathematically proven that a quintic polynomial with the applied method provided at least fourth-order convergence, which was further confirmed by numerical experiments. The experiments also showed that energy was conserved despite structured, bounded fluctuations, as expected for a variational integrator. Stability was not addressed.

Quartic Hermite time interpolation was applied in the structural dynamics context by Razavi *et al.* [109] and compared to members of the Newmark family. The derivation assumed constant mass, damping, and stiffness matrices, so it applied only to linear systems. The external load was assumed to vary linearly within the time step. The unique solution was defined by satisfaction of equilibrium equation at the beginning of the time step, at the end of the time step, and on average over the time step (*i.e.*, the time integral of the residual vanishes). The method was labeled a weighted residual method due to the use of the residual equation as an eliminant. Numerical experiments using a linear, unforced oscillator corroborated the fourth-order convergence rate of [81]. The response also showed no dissipation and minimal period distortion. Two regions of instability were stated in approximate terms: $0.51 < \Delta t/T < 0.55$ and $\Delta t/T > 1.24$, where the dimensionless sampling frequency $\Delta t/T$ is the time step size divided by the signal's period T.

Summary

In this chapter, the unique characteristics of membranes have been introduced, and numerous techniques for predicting their behavior have been summarized. The discussion illustrates why certain techniques were selected for the current research. For example, spring-mass models were rejected because they clearly have significant difficulty representing continuous structures. The linear membrane model enabled investigation of the simultaneous time-continuous Galerkin technique because it required fewer degrees of freedom than a nonlinear model, a key consideration because the method produces an extremely large linear system to solve. Likewise, the discontinuous Galerkin method was rejected because it increases the number of degrees of freedom. For the nonlinear membrane model, the point collocation spatial discretization combined with the group FE formulation potentially provided simplicity, efficiency, and accuracy. The Hermite time interpolation method was anticipated to offer higher accuracy with acceptable stability limits. Combining point collocation and Hermite interpolation into a dynamic membrane model results in a purely collocation-based scheme, a unique and promising formulation for a membrane problem.

III. Linear Analysis of Membrane Dynamics

This chapter details the development of a space-time finite element (FE) model to simulate a linear membrane. Space-time FE formulations of non-periodic systems typically utilize a time-marching algorithm. The solution is obtained for one slab of elements, where a slab is defined as the set of elements in a given time increment t_0 to $t_1 = t_0 + \Delta t$. The nodal values at t_1 are then used as initial conditions for the following time increment.

The goal of this portion of the research was to investigate a space-time FE method simultaneously discretized and solved across the entire space-time domain. This approach results in extremely large linear systems. Because the discontinuous Galerkin methods with low-order interpolations require more degrees of freedom for the same mesh [145], continuous interpolation functions in both space and time (trilinear elements) are employed. Thus, using the terminology of [62], it is labeled a time-continuous Galerkin (TCG) method. Further, the proposed approach will be referred to as the simultaneous time-continuous Galerkin (STCG) method to distinguish it from time-marching methods.

After deriving the governing equations for a mixed formulation, stability will be examined through numerical experimentation, and static and dynamic verification will be used to assess the accuracy of the scheme.

Membrane Model

The linear membrane model described in Section II is employed, which classically leads to the linear, second-order PDE of Eq. 1. With displacement $w(x, y, t)$, specific density $\bar{\rho}(x, y)$, and tension per unit length $P(x, y)$, the kinetic energy density (the kinetic energy per unit area, or over the infinitesimal spatial area $d\Omega$ where Ω is the spatial domain) is

$$\mathcal{T}_{d\Omega} = \frac{1}{2}\bar{\rho}\dot{w}^2 \tag{17}$$

The strain energy density is a function of the membrane gradients and the tension.

$$\mathcal{U}_{d\Omega} = \frac{1}{2}P\left[\left(\frac{\partial w}{\partial x}\right)^2 + \left(\frac{\partial w}{\partial y}\right)^2\right] \tag{18}$$

Mixed Form.

Discretization of the governing equation in its current form results in a displacement formulation. To obtain a mixed formulation, the second-order equation will be resolved into a system of first-order equations by the application of constraints. Using the variational approach, the governing equations will now be re-derived using Lagrange Multipliers to introduce new dependent variables and enforce the constraints.

The generalized momentum p and generalized stresses σ are defined as

$$p \equiv \frac{\partial \mathcal{T}_{d\Omega}}{\partial \dot{w}} = \bar{\rho}\frac{\partial w}{\partial t} \tag{19}$$

$$\sigma_x \equiv \frac{\partial \mathcal{U}_{d\Omega}}{\partial w/\partial x} = P\frac{\partial w}{\partial x} \tag{20}$$

$$\sigma_y \equiv \frac{\partial \mathcal{U}_{d\Omega}}{\partial w/\partial y} = P\frac{\partial w}{\partial y} \tag{21}$$

The distinction that these terms are “generalized” indicates the variables are not necessarily based upon a physical model. In particular, the generalized stresses should not be confused with Cauchy stresses. These relationships will be enforced by the

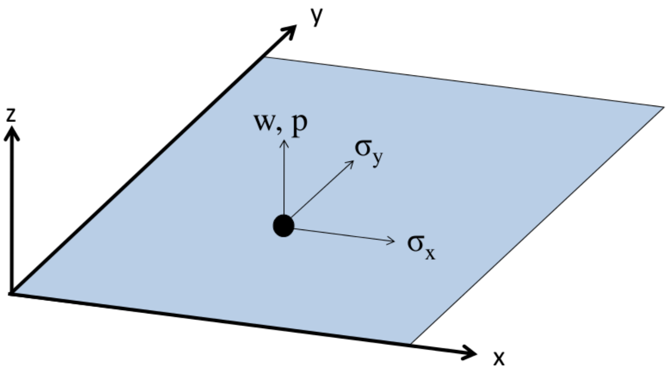


Figure 1. Diagram of degrees of freedom for the mixed formulation.

constraint equations $g_i = 0$, where

$$g_1 = p - \bar{\rho}\dot{w} \tag{22}$$

$$g_2 = \sigma_x - P\frac{\partial w}{\partial x} \tag{23}$$

$$g_3 = \sigma_y - P\frac{\partial w}{\partial y} \tag{24}$$

Taking the variation of the constraint equations for use later in the derivation, we obtain

$$\delta g_1 = \delta p - \bar{\rho}\frac{\partial \delta w}{\partial t} \tag{25}$$

$$\delta g_2 = \delta\sigma_x - P\frac{\partial \delta w}{\partial x} \tag{26}$$

$$\delta g_3 = \delta\sigma_y - P\frac{\partial \delta w}{\partial y} \tag{27}$$

In terms of the generalized momentum, the variation of the kinetic energy expression becomes

$$\delta\mathcal{T}_{d\Omega} = \bar{\rho}\dot{w}\,\delta\dot{w} = \frac{1}{\bar{\rho}}p\delta p \tag{28}$$

Similarly, in terms of generalized strains, the potential energy becomes

$$\mathcal{U}_{d\Omega} = \frac{1}{2P}\left(\sigma_x^2 + \sigma_y^2\right) \tag{29}$$

and its variation is

$$\delta\mathcal{U}_{d\Omega} = \frac{1}{P}\left(\sigma_x\delta\sigma_x + \sigma_y\delta\sigma_y\right) \tag{30}$$

The virtual work due to a prescribed external pressure force f, where in the linear membrane case $f(x, y, t)$ acts in the same direction as the displacement w, is expressed as

$$\overline{\delta\mathcal{W}}_{d\Omega} = f\delta w \tag{31}$$

Applying Hamilton's Law of Varying Action.

To apply Hamilton's Law of Varying Action, the energy functional is formed .

$$I = \int_{t_0}^{t_1}\left(\mathcal{T} - \mathcal{U} + \sum_i \lambda_i g_i\right) dt \tag{32}$$

Next, the sum of virtual work and the first variation of the energy functional is set to zero [7].

$$\delta I + \overline{\delta\mathcal{W}} = 0 \tag{33}$$

The expressions for the individual terms are substituted, and the chain rule is applied to the Lagrange Multiplier summations, leading to the expanded expression

$$0 = \int_{t_0}^{t_1}\iint_{\Omega}\left[\frac{p}{\bar{\rho}}\delta p - \frac{\sigma_x}{P}\delta\sigma_x - \frac{\sigma_y}{P}\delta\sigma_y + \sum_i \lambda_i\delta g_i + \sum_i g_i\delta\lambda_i + f\delta w\right] d\Omega dt \tag{34}$$

Substitution of the constraint functions g_i results in the functional

$$\begin{aligned} 0 \;=\; & \int\limits_{t_0}^{t_1} \iint\limits_{\Omega} \Bigg[\frac{p}{\bar{\rho}} \delta p - \frac{\sigma_x}{P} \delta\sigma_x - \frac{\sigma_y}{P} \delta\sigma_y + f\delta w \\ & + \lambda_1 \left(\delta p - \bar{\rho} \frac{\partial \delta w}{\partial t} \right) + \lambda_2 \left(\delta\sigma_x - P \frac{\partial \delta w}{\partial x} \right) + \lambda_3 \left(\delta\sigma_y - P \frac{\partial \delta w}{\partial y} \right) \\ & + \left(p - \bar{\rho} \frac{\partial w}{\partial t} \right) \delta\lambda_1 + \left(\sigma_x - P \frac{\partial w}{\partial x} \right) \delta\lambda_2 + \left(\sigma_y - P \frac{\partial w}{\partial y} \right) \delta\lambda_3 \Bigg] d\Omega dt \end{aligned} \tag{35}$$

Integration by parts is applied to each term that contains a derivative of a variation of a variable. This procedure moves the derivatives to the Lagrange multipliers and extracts the boundary conditions.

$$-\int\limits_{t_0}^{t_1} \iint\limits_{\Omega} \bar{\rho}\lambda_1 \frac{\partial \delta w}{\partial t}\, d\Omega dt = \int\limits_{t_0}^{t_1} \iint\limits_{\Omega} \bar{\rho} \frac{\partial \lambda_1}{\partial t} \delta w \, d\Omega dt - \iint\limits_{\Omega} \bar{\rho}\lambda_1 \, \delta w \, d\Omega \, \Big|_{t_0}^{t_1} \tag{36}$$

$$-\int\limits_{t_0}^{t_1} \iint\limits_{\Omega} P\lambda_2 \frac{\partial \delta w}{\partial x}\, d\Omega dt = \int\limits_{t_0}^{t_1} \iint\limits_{\Omega} P \frac{\partial \lambda_2}{\partial x} \delta w \, d\Omega dt - \int\limits_{t_0}^{t_1} \int\limits_{\Gamma} P\lambda_2 \, \delta w \, n_x \, d\Gamma dt \tag{37}$$

$$-\int\limits_{t_0}^{t_1} \iint\limits_{\Omega} P\lambda_3 \frac{\partial \delta w}{\partial y}\, d\Omega dt = \int\limits_{t_0}^{t_1} \iint\limits_{\Omega} P \frac{\partial \lambda_3}{\partial y} \delta w y \, d\Omega dt - \int\limits_{t_0}^{t_1} \int\limits_{\Gamma} P\lambda_3 \, \delta w \, n_y \, d\Gamma dt \tag{38}$$

After performing the integration by parts, the functional is

$$\begin{aligned}
0 \;=\; & \int\limits_{t_0}^{t_1} \iint\limits_{\Omega} \left[\frac{p}{\bar{\rho}}\delta p - \frac{\sigma_x}{P}\delta\sigma_x - \frac{\sigma_y}{P}\delta\sigma_y + f\delta w \right. \\
& + \lambda_1\delta p + \lambda_2\delta\sigma_x + \lambda_3\sigma_y + \bar{\rho}\frac{\partial\lambda_1}{\partial t}\delta w + P\frac{\partial\lambda_2}{\partial x}\delta w + P\frac{\partial\lambda_3}{\partial y}\delta w \\
& \left. + \left(p - \bar{\rho}\frac{\partial w}{\partial t}\right)\delta\lambda_1 + \left(\sigma_x - P\frac{\partial w}{\partial x}\right)\delta\lambda_2 + \left(\sigma_y - P\frac{\partial w}{\partial y}\right)\delta\lambda_3 \right] d\Omega dt \\
& - \iint\limits_{\Omega} \bar{\rho}\lambda_1\,\delta w\,d\Omega\,\Big|_{t_0}^{t_1} \\
& - \int\limits_{t_0}^{t_1}\int\limits_{\Gamma} \left[P\lambda_2\,\delta w\,n_x + P\lambda_3\delta w\,n_y\right] d\Gamma dt \qquad (39)
\end{aligned}$$

Next we collect terms associated with each variational term.

$$\begin{aligned}
0 \;=\; & \int\limits_{t_0}^{t_1} \iint\limits_{\Omega} \left[\left(\bar{\rho}\frac{\partial\lambda_1}{\partial t} + P\frac{\partial\lambda_2}{\partial x} + P\frac{\partial\lambda_3}{\partial y} + f\right)\delta w \right. \\
& + \left(\lambda_1 + \frac{p}{\bar{\rho}}\right)\delta p + \left(\lambda_2 - \frac{\sigma_x}{P}\right)\delta\sigma_x + \left(\lambda_3 - \frac{\sigma_y}{P}\right)\delta\sigma_y \\
& \left. + \left(p - \bar{\rho}\frac{\partial w}{\partial t}\right)\delta\lambda_1 + \left(\sigma_x - P\frac{\partial w}{\partial x}\right)\delta\lambda_2 + \left(\sigma_y - P\frac{\partial w}{\partial y}\right)\delta\lambda_3 \right] d\Omega dt \\
& - \iint\limits_{\Omega} \bar{\rho}\lambda_1\,\delta w\,d\Omega\,\Big|_{t_0}^{t_1} \\
& - \int\limits_{t_0}^{t_1}\int\limits_{\Gamma} \left[P\lambda_2\,\delta w\,n_x + P\lambda_3\delta w\,n_y\right] d\Gamma dt \qquad (40)
\end{aligned}$$

Since this equation must be true for all permissible functions δw, δp, $\delta\sigma_x$, and $\delta\sigma_y$, the meanings of the Lagrange multipliers can be determined by inspection.

$$\lambda_1 = -\frac{p}{\bar{\rho}} \tag{41}$$

$$\lambda_2 = \frac{\sigma_x}{P} \tag{42}$$

$$\lambda_3 = \frac{\sigma_y}{P} \tag{43}$$

Substituting these expressions into the functional eliminates the Lagrange multipliers.

$$\begin{aligned} 0 \;=\; & \int_{t_0}^{t_1} \iint_{\Omega} \left[\left(-\frac{\partial p}{\partial t} + \frac{\partial \sigma_x}{\partial x} + \frac{\partial \sigma_y}{\partial y} + f \right) \delta w \right. \\ & \left. - \left(p - \bar{\rho}\frac{\partial w}{\partial t} \right) \frac{\delta p}{\bar{\rho}} + \left(\sigma_x - P\frac{\partial w}{\partial x} \right) \frac{\delta \sigma_x}{P} + \left(\sigma_y - P\frac{\partial w}{\partial y} \right) \frac{\delta \sigma_y}{P} \right] d\Omega dt \\ & + \iint_{\Omega} p\, \delta w \, d\Omega \, \Big|_{t_0}^{t_1} \\ & - \int_{t_0}^{t_1} \int_{\Gamma} \left[\sigma_x \, \delta w \, n_x + \sigma_y \delta w \, n_y \right] d\Gamma dt \end{aligned} \tag{44}$$

By collecting the terms according to their variational components, we obtain the mixed form of the governing equation and recover the three constraint equations. These four equations collectively form the basis for the upcoming finite element development.

$$\frac{\partial p}{\partial t} - \frac{\partial \sigma_x}{\partial x} - \frac{\partial \sigma_y}{\partial y} = f \tag{45}$$

$$p - \bar{\rho}\frac{\partial w}{\partial t} = 0 \tag{46}$$

$$\sigma_x - P\frac{\partial w}{\partial x} = 0 \tag{47}$$

$$\sigma_y - P\frac{\partial w}{\partial y} = 0 \tag{48}$$

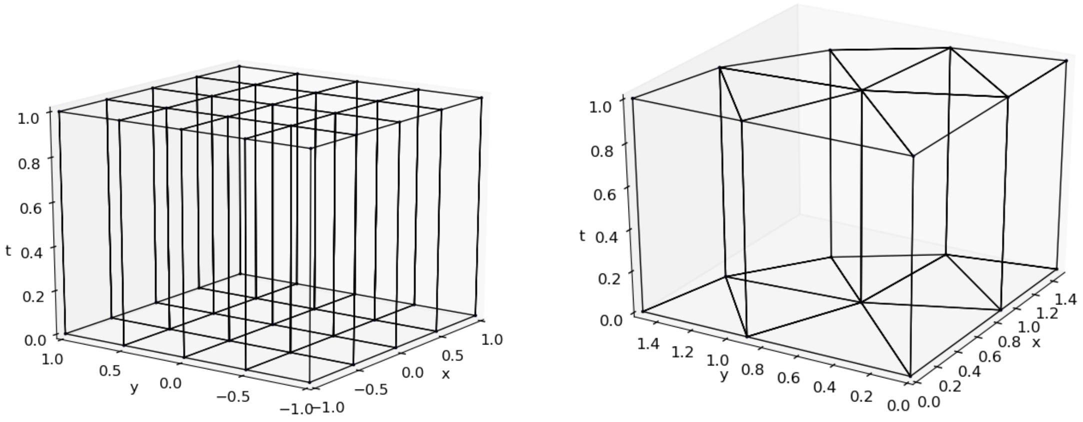
Domain Discretization

The space-time mesh was constructed by duplicating the spatial mesh at each time increment $t + n\Delta t$. Each initial spatial node (x, y, t_0), referred to in this paper as a parent node, has the offspring $(x, y, t_0 + n\Delta t)$. Thus each node in the domain is uniquely identified by its (x, y, t) coordinates. For this study Δt was constant, though it could certainly be varied to optimize performance. Each node has four degrees of freedom: w, p, σ_x, and σ_y, any of which could be fixed or otherwise prescribed.

Elements.

Two well-known three-dimensional solid elements were utilized as space-time elements. Both are basic trilinear isoparametric elements. Using ξ and η to denote in-plane local coordinates, the local coordinate system is denoted as (ξ, η, τ) and is aligned with the global system (x, y, t). Though the third axis is time rather than space, no modifications to the interpolation functions or integration procedures were necessary. Conceptually the resulting elements may be thought of as two planar membrane elements, one at time t and the other at time $t + \Delta t$, stacked upon each other to create an internal volume that is a space-time subdomain $\mathcal{D}$. A slab is defined as the set of all of the elements in space between two given times [60], so the entire solution domain consists of a stack of N_t slabs where N_t is the number of time steps. In this paper, the term "surface element" refers to a quadrilateral or triangle face in the (x, y) plane that is shared between two elements at the given time slice.

The first element is variously called a trilinear hexahedral [58], hexahedron [102], or when using right angles a linear brick [58]. It is an extension of the bilinear quadrilateral into the third dimension. The complete interpolation polynomial is $N_i = (1 + \xi_i\xi)(1 + \eta_i\eta)(1 + \tau_i\tau)/8$ where $\xi, \eta, \tau \in [-1, 1]$, providing C^0 continuity between elements. The 4th-order accurate 2x2x2 integration points are $(\xi, \eta, \tau) =$



(a) Hexahedra

(b) Triangular prisms

Figure 2. Representative slabs of elements.

$(\pm\sqrt{1/3}), (\pm\sqrt{1/3}), (\pm\sqrt{1/3})$. Reduced integration performed poorly and was not used.

The second element is the triangular prism, also called a wedge or a right pentahedron [102]. It can be viewed as an extension of the constant-strain triangle or a degenerate case of the hexahedron [58]. The C^0 linear shape functions are given in Eq. 49 [67] with $\xi, \eta \in [0, 1]$ and $\tau \in [-1, 1]$.

$$\begin{aligned} N_i &= \frac{1}{2}(1 - \xi - \eta)(1 + \tau_i \tau), \quad i = 1, 4 \\ N_i &= \frac{1}{2}\xi(1 + \tau_i \tau), \quad i = 2, 5 \\ N_i &= \frac{1}{2}\eta(1 + \tau_i \tau), \quad i = 3, 6 \end{aligned} \tag{49}$$

The 3x2 integration points are $(\xi, \eta) = (\frac{1}{6}, \frac{1}{6}), (\frac{2}{3}, \frac{1}{6}), (\frac{1}{6}, \frac{2}{3})$ at $\tau = \pm\sqrt{1/3}$, sufficient for up to a second degree polynomial. As with the hexahedral element, reduction of the integration order severely degraded the results.

The local element matrices were constructed to satisfy the governing equations as shown in Eq. 50. Since there are four degrees of freedom per node the resulting block stiffness matrix $\mathbf{K}$ was 32x32 for a hexahedron and 24x24 for a triangular prism. Boundary conditions were imposed locally by removing the appropriate rows and columns from the stiffness matrix and adjusting the right-hand side accordingly [28].

$$\begin{bmatrix} \mathbf{0} & \mathbf{K}_{vvt} & -\mathbf{K}_{vvx} & -\mathbf{K}_{vvy} \\ -\bar{\rho}\mathbf{K}_{vvt} & \mathbf{K}_{vv} & \mathbf{0} & \mathbf{0} \\ -P\mathbf{K}_{vvx} & \mathbf{0} & \mathbf{K}_{vv} & \mathbf{0} \\ -P\mathbf{K}_{vvy} & \mathbf{0} & \mathbf{0} & \mathbf{K}_{vv} \end{bmatrix} \begin{Bmatrix} \mathbf{w} \\ \mathbf{p} \\ \sigma_x \\ \sigma_y \end{Bmatrix} = \begin{Bmatrix} \mathbf{K}_{vv}\mathbf{f} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix} \tag{50}$$

$$\mathbf{K}_{vv} = \int_{\mathcal{D}} \mathbf{N}^T \mathbf{N} J d\mathcal{D} \tag{51}$$

$$\mathbf{K}_{vvt} = \int_{\mathcal{D}} \mathbf{N}^T \frac{\partial \mathbf{N}}{\partial t} J d\mathcal{D} \tag{52}$$

$$\mathbf{K}_{vvx} = \int_{\mathcal{D}} \mathbf{N}^T \frac{\partial \mathbf{N}}{\partial x} J d\mathcal{D} \tag{53}$$

$$\mathbf{K}_{vvy} = \int_{\mathcal{D}} \mathbf{N}^T \frac{\partial \mathbf{N}}{\partial y} J d\mathcal{D} \tag{54}$$

where $\mathcal{D}$ is the element subdomain of the problem domain $\Omega \times [t_0, t_f]$. The non-zero values were then inserted into the sparse, square global stiffness matrix. For a discretization with N_s spatial nodes and N_t time steps, and accounting for the four degrees of freedom per node due to the mixed formulation, the matrix size is $[4N_s(N_t + 1)]^2$ minus rows and columns eliminated through imposition of boundary or initial conditions. For dense spatial meshes or long simulations, hardware capacity clearly becomes a practical concern and at some point this method becomes unsuitable. The formulation results in a single linear equation, the solution of which provides the displacement, momentum, and in-place stresses for each node.

After obtaining the solution vector, post-processing matched the degrees of freedom to the parent nodes to create the response history for each surface node of the membrane.

Static Verification and Spatial Convergence.

For the present method, a static problem is a special case of the general method – the same algorithm is used for both static and dynamic analyses. For a static analysis the generalized momentum p is set to zero for every node in the space-time mesh, and spatial boundary conditions are fixed to their values at $t = 0$. These constraints are sufficient to determine a unique static solution. Though only one time step is required to generate a single slab of elements and subsequent static solution, the solution is identical for an arbitrarily large number of time steps. The equilibrium of each static solution was verified during post-processing by checking that the field values of a parent node were identical to those of its offspring.

Static verification was performed by use of Examples 2.8 and 4.1 in Reddy's textbook [110], which conveniently provide both an analytical solution and finite element approximations. The FE solutions were obtained using displacement formulation and bilinear quadrilateral and triangle elements on two grids (element size h^e= 0.25 and $h^e = 0.125$, or equivalently as the number of elements in his mesh, $N = 4$ and $N = 8$). Setting $P = \bar{\rho} = f = 1$, the governing equation is the Poisson equation,

$$-\nabla^2 w = 1 \tag{55}$$

The domain was a 1×1 square, $x, y \in [0, 1]$, with mixed boundary conditions

$$w(x, 1) = 0 \tag{56}$$

$$w(1, y) = 0 \tag{57}$$

$$\frac{\partial w}{\partial x}(0, y) = 0 \tag{58}$$

$$\frac{\partial w}{\partial y}(x, 0) = 0 \tag{59}$$

The series solution was therefore

$$w(x, y) = \frac{1}{2}\left\{\left(1 - y^2\right) + \frac{32}{\pi^3}\sum_{n=1}^{\infty}\frac{(-1)^n \cos\left[(2n-1)\,\pi y/2\right]\cosh\left[(2n-1)\,\pi x/2\right]}{(2n-1)^3\cosh\left(2n-1\right)\pi/2}\right\} \tag{60}$$

The STCG method was employed using both types of elements. The error, ϵ, was defined as the displacement error at the origin. The observed rate of convergence for both elements was two, as shown in Figure 4. The errors were approximately the same as those of Reddy's displacement method.

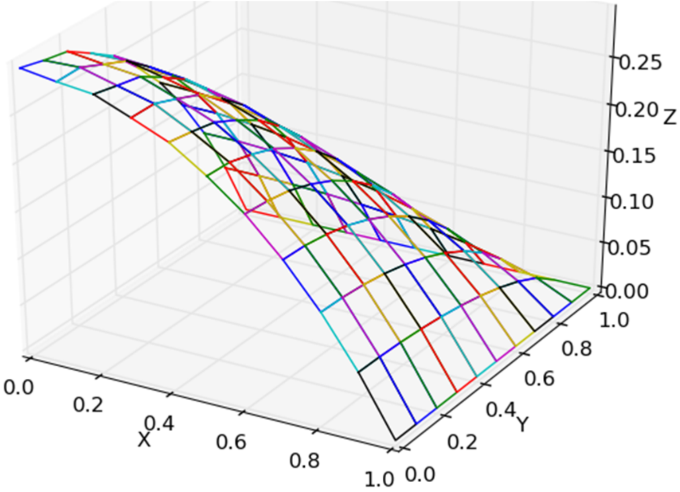


Figure 3. Representative Poisson equation solution using square elements ($h^e = 0.1$).

Figure 5 plots STCG results on the x-axis for the two coarsest mesh sizes (the same as used by Reddy). Mild variations of the solution are evident, a phenomenon that can occur in mixed formulations with certain interpolation combinations [19, 146]. In this case they reduced significantly as the mesh was refined and the model remained consistent.

Time Integration Performance.

In this section we investigate the accuracy and reliability of the model in reproducing periodic signals of varying frequency. The objective is to derive a methodology

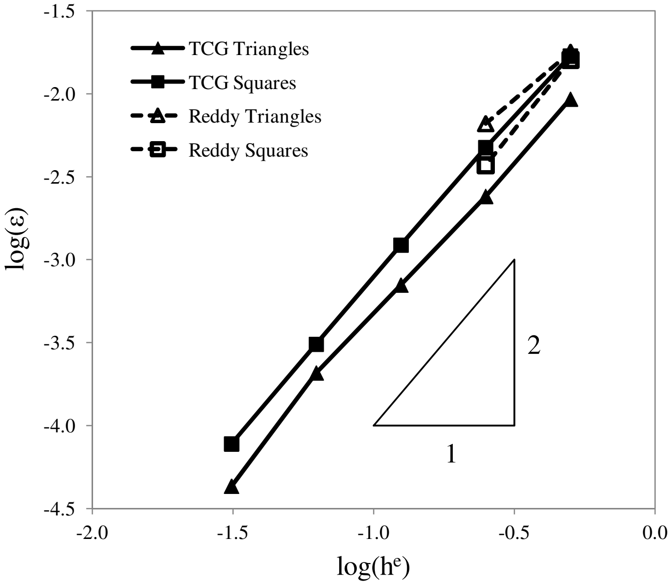
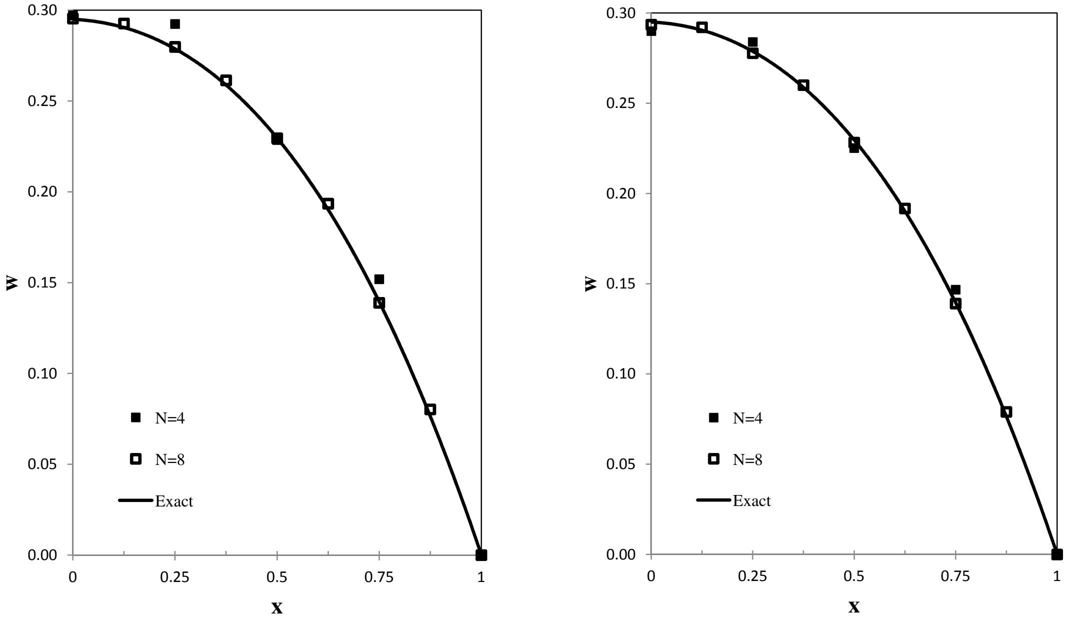


Figure 4. Convergence of the center displacement for both element types.



(a) Triangle elements.

(b) Square elements.

Figure 5. Solutions along the x-axis for the static Poisson problem.

for determining Δt and characterizing limitations and restrictions on the choice. At least ten samples per period are required by rule of thumb [120] and as the practical limit for a linear approximation of a periodic signal [130]. These guidelines as well as stability limits will be tested by numerical experimentation and compared to other methods. Let ω be the angular frequency. Frequencies will be normalized by the sampling rate ω_s. Depending on context, the normalized frequency will be written either as ω^* where $\omega^* = \omega/\omega_s$, or as $\Delta t/T$, which is numerically identical and found in finite element texts [8, 58]. To avoid high frequencies being folded into the lower frequency band (aliasing) one must sample at more than twice the Nyquist frequency ($\omega_s > 2\omega_N$, where $\omega_N^* = 0.5$) [105, 33, 120]. Finally, since signals are distorted by the model, frequencies observed in model output are denoted with a bar (e.g., $\bar{\omega}^*$). It should be noted that a spectrum of normalized frequencies is present in every dynamic response signal.

First we look at how the sampling rate affects the Gaussian quadrature accuracy in the time axis. Consider the time span of one element slab Δt and a periodic signal of period T traversing it. The fraction of the period covered by the element has already been defined as the normalized frequency $\omega^* = \Delta t/T$. Consider also that the element may be sampling any portion of the signal depending on the phase ϕ. The error is the percentage difference between the results of the well-known two-point Gaussian quadrature integration and the analytical solution. The bounds of the integration error shown in Figure 6 depict the normalized error at the worst-case ϕ. As the sampling rate approaches infinity ($\Delta t/T \rightarrow 0$), the error approaches zero because the linear approximation approaches the exact solution regardless of the phase. Also regardless of phase, when sampling at the Nyquist frequency the linear quadrature produces a zero result (the integration points will be equidistant from the t-axis and on opposite sides). The error is less than 3% for $\omega^* = 0.1$ and less than 13% for

$\omega^* = 0.2$.

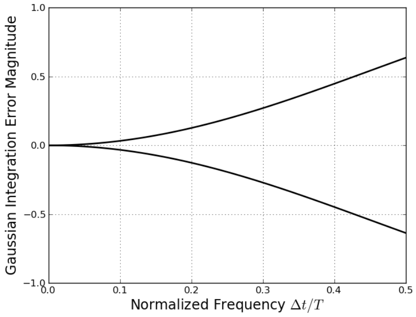


Figure 6. Worst-case quadrature error bounds for all phase angles.

For the following numerical investigations a single degree-of-freedom simple harmonic oscillator was formed using two equally-sized surface elements in the (x, y) plane. Their edges were joined along the axis $x = 0$, they were placed between simple supports located at $x = \pm 1$, subjected to an initial displacement $w(0, y, 0) = 1$ along their interface, and released. The solution was $w = \cos \omega t$ where $\omega = 1.5c/h$.

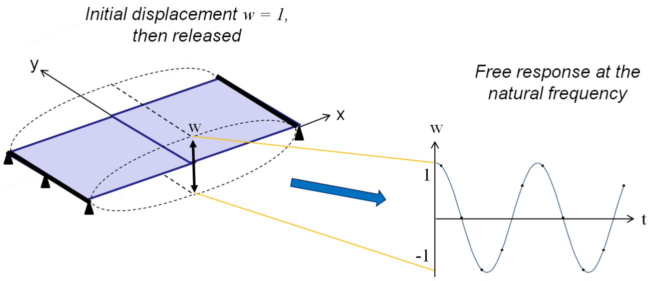

Initial displacement w = 1, then released
y
x
w
Free response at the natural frequency
w
1
-1
t

Figure 7. Diagram of the simple harmonic oscillator configuration.

First a convergence study in time was performed. The simulation was run for two periods with varying numbers of elements (equivalent to varying the normalized frequency). The error was defined as the mean error magnitude for all solution points. As shown in Figure 8, the observed rate of convergence was two.

Next the oscillator was run for 100 periods over a series of runs with different values of ω^* for the oscillator. The values of ω^* were varied by changing the oscillator's

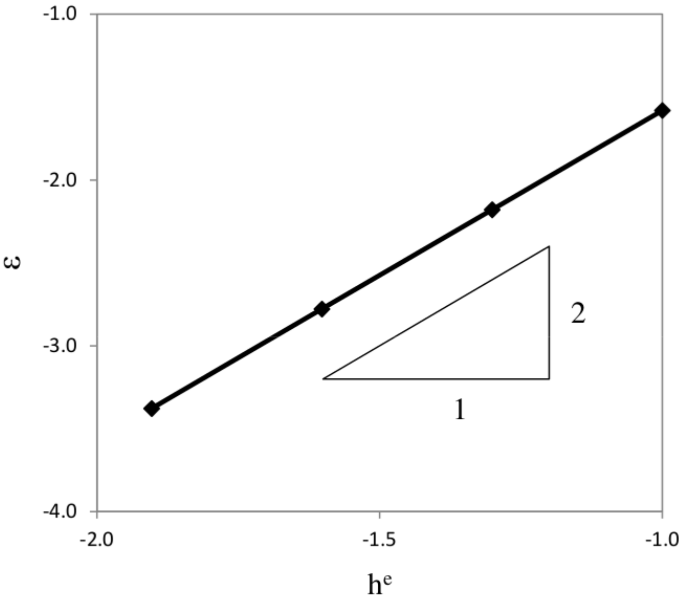


Figure 8. Time convergence results for the STCG model.

natural frequency with E. As seen in Figure 9, a spurious mode was evident at a higher frequency than the fundamental response. As ω^* moves right from the origin the spurious mode migrates leftward from ω_N. This behavior often indicates the spurious peak has been aliased. The frequencies of the peaks given the known input frequency are shown in Figure 10(a). Given an ω^* on the vertical axis, one reads horizontally right to find the model-predicted normalized frequencies $\bar{\omega}^*$ of both responses. In every case the peaks merged when $\omega^* \approx 0.276$ and $\bar{\omega}^* \approx 1/3$. The amplitude distortion as the frequency increases may be seen in Figure 10(b) where the largest observed displacement and the mean of the displacement magnitudes for the entire run are depicted for various runs. For a perfectly reproduced signal, the maximum would be 1.0 and the average magnitude approximately 0.64. A ramp-up of the maximum observed displacement was evident prior to the peaks merging. When the frequency of the oscillator was raised further the solution was severely damped and converged to zero within a few times steps. Though the precise mechanism in this model has not yet been determined, this behavior has been noted in finite element complex wavenumber Fourier analyses [130]. Accordingly we will refer to $\bar{\omega}^* \approx 0.333$ as the cutoff frequency.

Nearly identical behavior was mathematically detailed and reproduced by [130] while performing a finite element complex wavenumber Fourier analysis. The following briefly describes their findings for comparison to the present results. Linear element discretization of the reduced wave equation for a uniform elastic bar produces regions of the frequency domain called the passing band and the stopping band. They are separated by the cutoff frequency α_{max} using a different normalization $\alpha = \omega h/c$. When $\alpha < \alpha_{max}$ the imaginary component of the eigenvalue is zero and the wave propagates (passes); when $\alpha > \alpha_{max}$ the imaginary components produce rapid damping, as seen in our Figure 10(b). For a diagonalized mass matrix $\alpha_{max} = 2$, and the reference frequency α_{ref} is set at ten elements per wavelength ($\omega^* = 0.1$ in our nomenclature). Then we have from their development the ratio

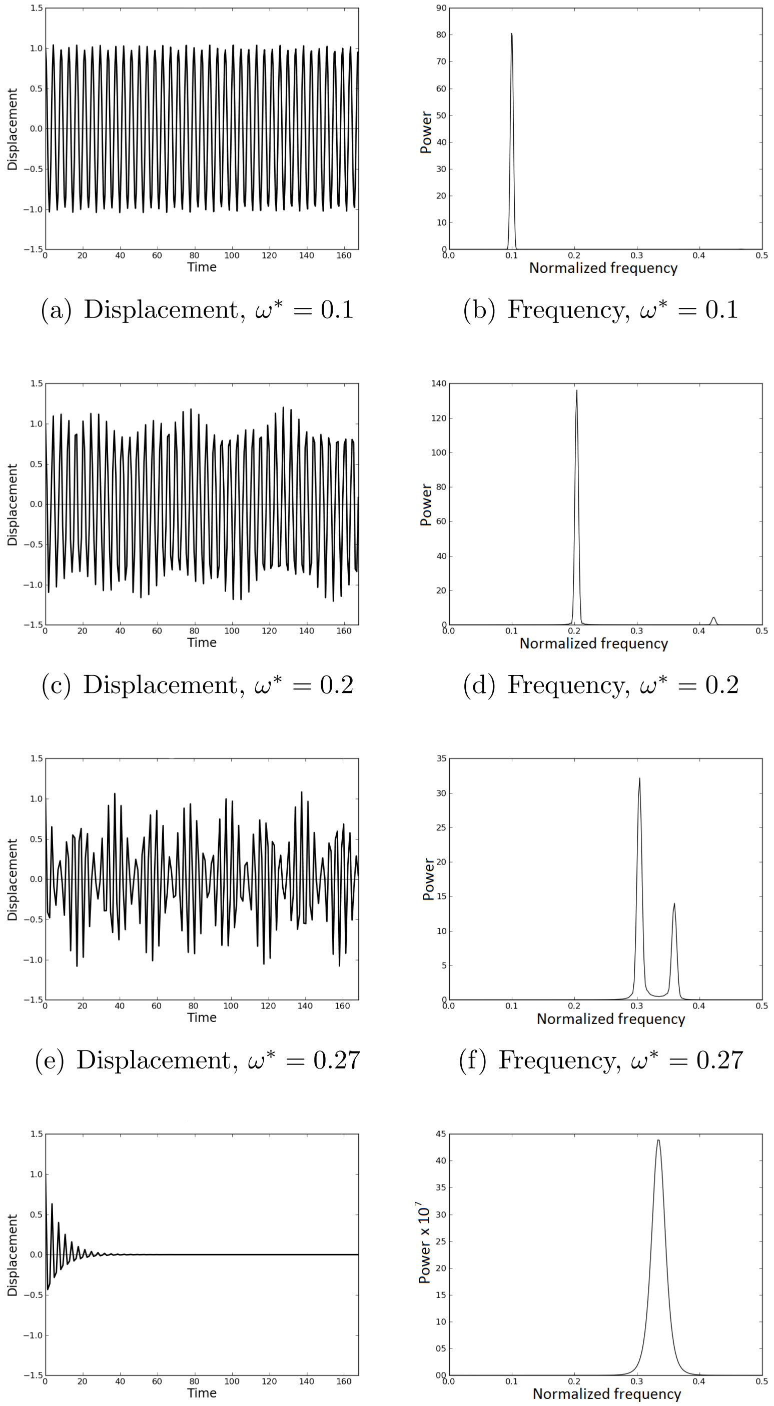
$$\frac{\alpha_{max}}{\alpha_{ref}} = \frac{2}{2\pi/10} = 10/\pi \tag{61}$$

Setting the ratio h/c constant, the ratio $\alpha_{max}/\alpha_{ref}$ is equal to the ratio of equivalent normalized frequencies in our development. Hence we can solve for our cutoff frequency. For the simple harmonic oscillator $\omega^* = 0.1$ and $\omega_s = 15$ rad/s produced an output signal at $\bar{\omega}^* = 1.52$ rad/s.

$$\bar{\omega}^*_{max} = \left(\frac{10}{\pi}\right)\bar{\omega}^*_{ref} = \left(\frac{10}{\pi}\right)\left(\frac{1.52}{15}\right) = 0.323 \tag{62}$$

This prediction is within 3% of our model's cutoff frequency. The comparison between results strongly suggests that the current technique is filtering out all content above $\omega^* \approx 0.276$.

The numerical dispersion in terms of period distortion was compared to that pro-



(a) Displacement, $\omega^* = 0.1$

(b) Frequency, $\omega^* = 0.1$

(c) Displacement, $\omega^* = 0.2$

(d) Frequency, $\omega^* = 0.2$

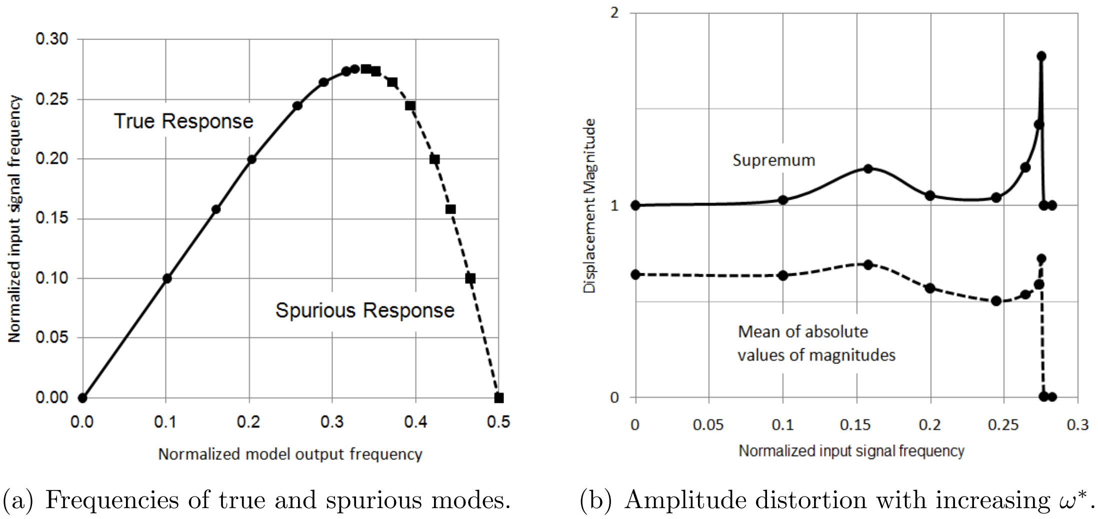
(e) Displacement, $\omega^* = 0.27$

(f) Frequency, $\omega^* = 0.27$

(g) Displacement, $\omega^* = 0.28$

(h) Frequency, $\omega^* = 0.28$

Figure 9. Displacement history and associated frequency response of the simple harmonic oscillator with increasing time steps.



(a) Frequencies of true and spurious modes. (b) Amplitude distortion with increasing ω^*.

Figure 10. Response frequency and magnitude behavior of the simple harmonic oscillator.

duced by two of the most commonly-used direct integration techniques: the central difference method and the trapezoidal rule. Both are sub-cases of the Newmark method. The central difference method is obtained when $\beta = 0$ and $\gamma = \frac{1}{2}$. It is explicit and therefore conditionally stable with the limit $\Delta t/T \leq 1/\pi$. The trapezoidal rule, where $\beta = \frac{1}{4}$ and $\gamma = \frac{1}{2}$, is symplectic with unconditional stability. The dispersion relationship for the present method can be seen on the left-hand curve of Figure 10(a), which shows a nearly linear relationship of $\omega^*/\bar{\omega}^*$ through $\omega^* = 0.2$. Separately, the one-dimensional central difference and trapezoid rule algorithms were used to solve the same ordinary differential equation [102]. The results are shown in Figure 11. As the normalized frequency increased, the trapezoid rule elongated the period and the central difference method compressed it, matching results in the literature [8]. The simultaneous TCG preserved the period to within 0.1% up to $\Delta t/T = 0.1$ and within 0.7% up to $\Delta t/T = 0.18$. This frequency preservation is vital since our primary goals include the reproduction of modal natural frequencies.

Although the present model was not susceptible to instability, the cutoff frequency places a clear limit on the useful part of the spectrum. From the cutoff frequency

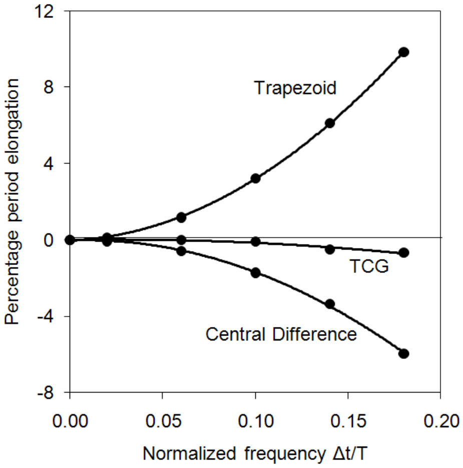


Figure 11. Period distortion comparison for different integration schemes

one can derive the maximum allowable time step Δt_{cr} in terms of element size and wave speed. Note that the standard notation h is used here for the representative maximum element spatial length, not to be confused with membrane thickness.

$$\Delta t_{cr} \approx 0.276T = 0.276\,(2\pi/\omega) = 0.276\frac{2\pi h}{1.5c} = 1.16\frac{h}{c} \tag{63}$$

The constant 1.16 represents the maximum allowable Courant Friedrichs-Lewy Number CFL. The cutoff frequency allows a slightly higher CFL than the broadly representative cases listed in Table 1 and is therefore less restrictive on the choice of Δt.

Table 1. Survey of representative values of CFL_{max} from the literature.

1.0	Wave equation with linear elements [58]
1.0	Unsupported axial bar using lumped masses [28]
0.707	Rectangular 4-node quadrilateral elements with central difference method [58]
0.577	Two-node linear rod with central difference method [58]

When solving the wave equation with a linear element and the Newmark method, $CFL = 1$ is not only limiting but also optimal [58]. Similarly, for a two-node linear rod element using the Newmark central difference method, the limit is $CFL = 0.577$.

$$\Delta t \leq \frac{h}{\sqrt{3}c} \approx 0.577\frac{h}{c} \tag{64}$$

Hughes also provided the results of studies of rectangular 4-node quadrilateral elements along with the central difference method, which resulted in $CFL = 0.707$:

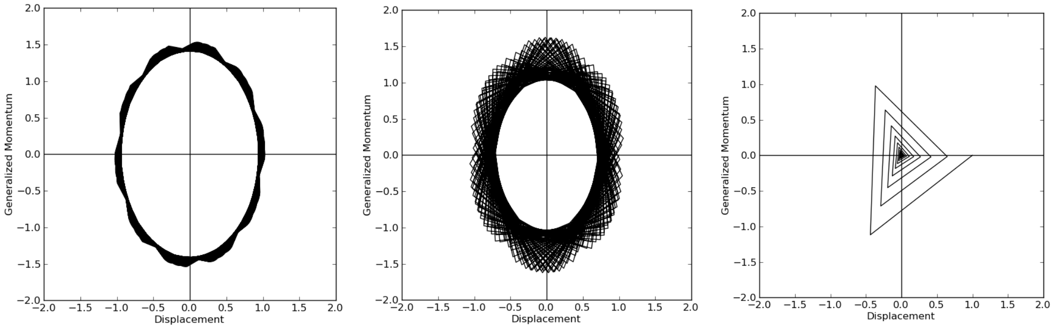
$$\Delta t \leq \frac{1}{c_d\left(1/h_1^2 + 1/h_2^2\right)^{1/2}} \approx 0.707\frac{h}{c} \tag{65}$$

where for generality we equalize the spatial dimensions $h \approx h_1 \approx h_2$ and recall that $c_d^2 = (\lambda + 2\mu)/\rho = E/\rho$ as found earlier. Lastly, Cook [28] modeled the mechanism as an unsupported axial bar of length h. The highest frequency is $\omega_{max} = \pi c/h$ since masses are not lumped. Thus he obtains $CFL = 0.637$ and the critical time step

$$\Delta t_{cr} \leq \frac{2}{\omega_{max}} = \frac{2h}{\pi c} \approx 0.637\frac{h}{c} \tag{66}$$

Conservation of energy was examined by running the simple harmonic oscillator for 100 periods. This time scale was sufficient for this study since the dynamic cases capture at most five periods of the first mode. The phase plots for oscillations at three different normalized frequencies are shown in Figure 12. At $\omega^* = 0.1$ the energy was effectively conserved. At $\omega^* = 0.2$, where integration error up to 13% has been demonstrated, the plot's slow migration towards the origin indicates a progressive loss of energy. At $\omega^* = 0.28$, just above the cutoff frequency, the extreme energy dissipation was obvious as the solution collapsed to the resting position. This phenomenon was also clear in terms of amplitude decay in Figure 10(b).

In this section we have established the following performance observations: frequencies are accurately reproduced through $\omega^* = 0.10$, accuracy degrades through



(a) $\omega^* = 0.1$ (b) $\omega^* = 0.2$ (c) $\omega^* = 0.28$

Figure 12. Energy conservation of the simple harmonic oscillator at different normalized natural frequencies.

$\omega^* = 0.20$ due to integration error, and results are increasingly unreliable as the cutoff frequency $\omega^* \approx 0.276$ is approached. Since the model aggressively dissipates signals above the cutoff frequency, frequency content there is noise. The next section outlines a methodology based on these results.

Membrane Dynamics

Cases and Methodology.

Two canonical cases were selected for comparing model results to analytical solutions: a rectangular membrane and a circular membrane. The setup parameters are provided in Table 2. Note that only one-fourth of the membrane was modeled in each case using zero-force boundary conditions. The steps of the method follow:

1. Choose the target frequency ω_{tgt}, the highest frequency that must be accurately reproduced.

2. Calculate the time increment. To ensure sufficient accuracy at ω_{tgt},

$$\Delta t = \frac{T_{tgt}}{10} = \frac{\pi}{5\omega_{tgt}} \tag{67}$$

3. Ensure the surface elements are properly sized (h is element size). From Eq. 63,

$$h \geq \frac{c\Delta t}{CFL_{max}} = 0.865 c\Delta t_{cr} \tag{68}$$

As noted in the literature for wave equation discretization [58, 28], CFL values close to 1.0 provided the best results. When CFL was near 0.5, for example, oscillations were introduced at wavelengths that interfered with the targeted modes. Thus, for best results strive for

$$h = c\Delta t \tag{69}$$

4. Run the simulation. All nodes at $t = 0$ require initial displacements and momenta.

5. Determine the highest valid frequency content in the model's results.

   (a) As demonstrated by the cutoff frequency, content with $T < 3\Delta t$ is not valid (the small wavelengths are not sampled at a sufficient rate).

   (b) When the CFL is low (element size large relative to the time increment), the oscillations span more time intervals. From Eq. 63 and the size limit in Eq. 68, the noise region is defined as $T < (4\pi/3)(h/c)$.

   After the results of the model have been obtained, one may consider invalid any content with a period of

$$T < max\left(3\Delta t, \frac{4\pi}{3}\frac{h}{c}\right) \tag{70}$$

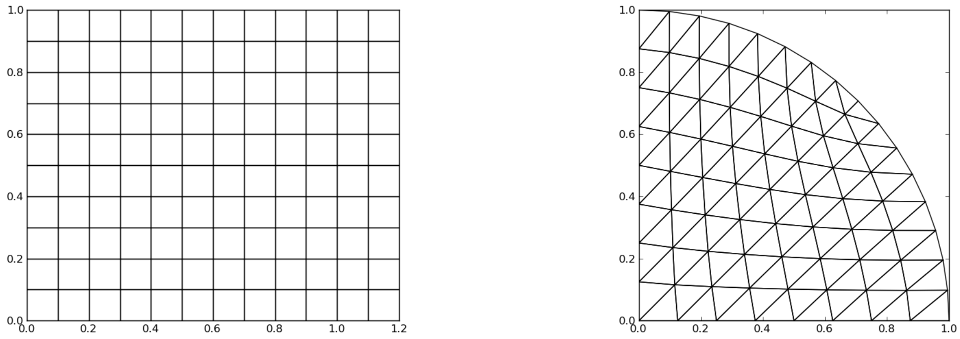
The analytical solution for the rectangular membrane is given in Eq. 71 [108]. To target the third modal frequency, Δt was calculated to be one tenth of the third mode's wavelength. The membrane was subjected to an initial velocity that was uniform across the entire surface.

Table 2. Dynamic test cases for the STCG model. Only one quadrant was modeled ($x > 0$ and $y > 0$).

	Rectangle	Circle
Forcing function	Uniform initial velocity	$r \leq 0.3a$: Impulse p_0 in $[t_0, t_1]$ $r > 0.3a$: Zero
Element type	Right Hexahedra	Triangular prisms
Dimensions	2.4 x 2.0	unit radius($a = 1$)
Surface mesh	143 nodes 120 elements	81 nodes 121 elements
Element length h	0.141	0.110
c	1.0	1.0
Δt	0.128	0.114
CFL	0.91	1.04
Stiffness matrix	$48,454$ square $2,305,821$ non-zero 0.10% non-zero	$31,424$ square $1,193,548$ non-zero 0.12% non-zero

$$\begin{aligned}
w_{mn}(x,y,t) &= \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} \sin\frac{m\pi x}{a}\sin\frac{n\pi y}{b} B_{mn}\sin\omega_{mn}t \\
\omega_{mn} &= c\pi\left[\left(\frac{m}{a}\right)^2 + \left(\frac{n}{b}\right)^2\right]^{1/2} \\
B_{mn} &= \frac{4}{ab\omega_{mn}}\int_0^a\int_0^b \dot{w}_0(x,y)\sin\frac{m\pi x}{a}\sin\frac{n\pi y}{b}dxdy \\
&= \frac{4}{mn\pi^2\omega_{mn}}(\cos m\pi - 1)(\cos n\pi - 1)
\end{aligned} \tag{71}$$

For the circular membrane the two-dimensional mesh as shown in Figure 13(b) was generated using the open source software Gmsh [43]. The membrane had an outer radius $a = 1$ and the impulse was a uniform pressure p_0 applied inside the radius $b = 0.3$ only at $t = 0$. The natural frequencies were $\omega_{mn} = c\beta_{mn}$ where $\beta_{mn}a$ are the zeros of the Bessel functions $J_m(\beta a)$. The exact solution is given in Eq. 72 [91]. The time increment was chosen to target the third modal frequency.



(a) Rectangular membrane.

(b) Circular membrane.

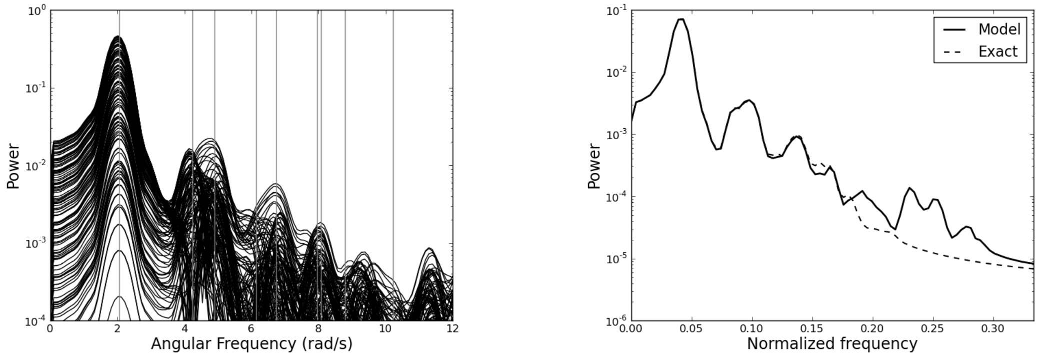
Figure 13. Meshes used for evaluation of the STCG method for the dynamic cases.

$$w(r,\theta,t) = \frac{2p_0bc}{\rho a^2}\sum_{n=1}^{\infty}\frac{J_1\left[(\omega_{0n}/c)\,b\right]J_0\left[(\omega_{0n}/c)\,r\right]}{\omega_{0n}^2 J_1^2\left[(\omega_{0n}/c)\,a\right]}\int_0^t f(\tau) sin\omega_{0n}\left(t-\tau\right)d\tau \qquad (72)$$

Modal Frequency Results.

The power spectral density (PSD) for the rectangular membrane with respect to $\bar{\omega}^*$ for every node is shown in Figure 14(a). The vertical lines indicate the exact solutions ω^* so any phase distortion can be discerned by offset peaks. The first seven mode frequencies were accurately predicted but beyond $\bar{\omega} = 8$ rad/s (approximate $\omega^* = 0.16$) the calculated modal frequencies are inaccurate. Figure 14(b) displays a single PSD that is the supremum of all nodal values at each frequency; thus all PSDs in Figure 14(a) lie under the single PSD in Figure 14(b). The dotted line indicates the exact solution for the discrete nodes and was thus equally affected by the sampling rate. The first seven modes were included in the exact solution.

The PSD plots for the circular membrane were created in the same manner as those

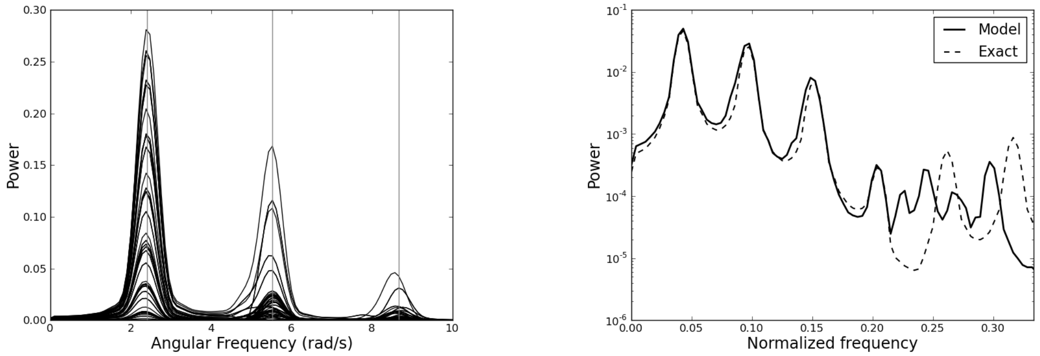


(a) Composite plot of PSDs for every node. Vertical lines depict exact modal frequencies.

(b) Comparison of model results to exact solution.

Figure 14. Frequency spectrum of the response of the rectangular linear membrane

of the rectangular membrane and are shown in Figure 15. The analytical solution in Figure 15(b) includes only the first four modes. The first three mode frequencies were accurately reproduced with the third mode at approximately $\omega^* = 0.15$. However, the spatially discontinuous impulse generated much more high-frequency noise than the uniform impulse in the rectangular membrane case. The noise above $\bar{\omega}^* = 0.25$ occluded the response.



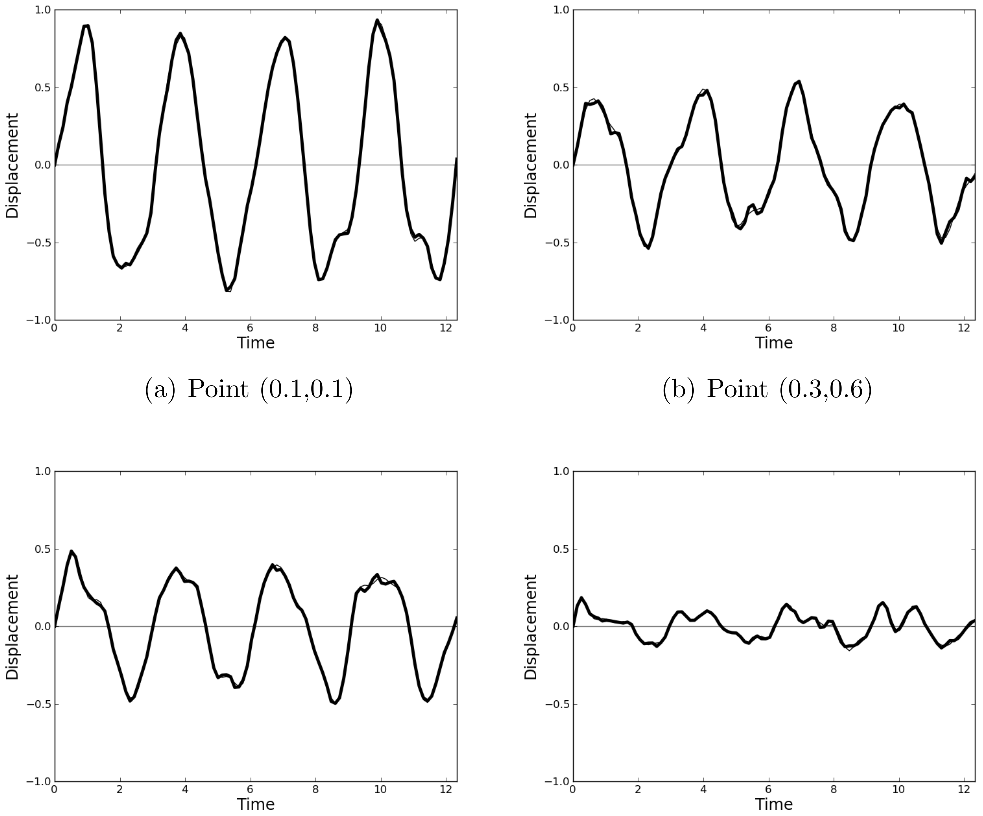
(a) Composite plot of PSDs for every node. Vertical lines depict exact modal frequencies.

(b) Comparison of model results to exact solution.

Figure 15. Frequency spectrum of the response of the circular linear membrane

Response History Results.

The response histories for four nodes of the rectangular membrane are shown in Figure 16. The thick line depicts the numerical prediction. The thin line depicts the analytical solution including the first seven modes. The axis limits for all of the plots are consistent to illustrate relative magnitudes. As expected from the PSD in Figure 14(b), high-frequency noise was present in the form of "jitters." Good examples can be seen in Figure 16(b) at four locations: the first two peaks, at 5.5 seconds, and at 12 seconds. Otherwise the model predictions closely matched the exact solution.



(a) Point (0.1,0.1)

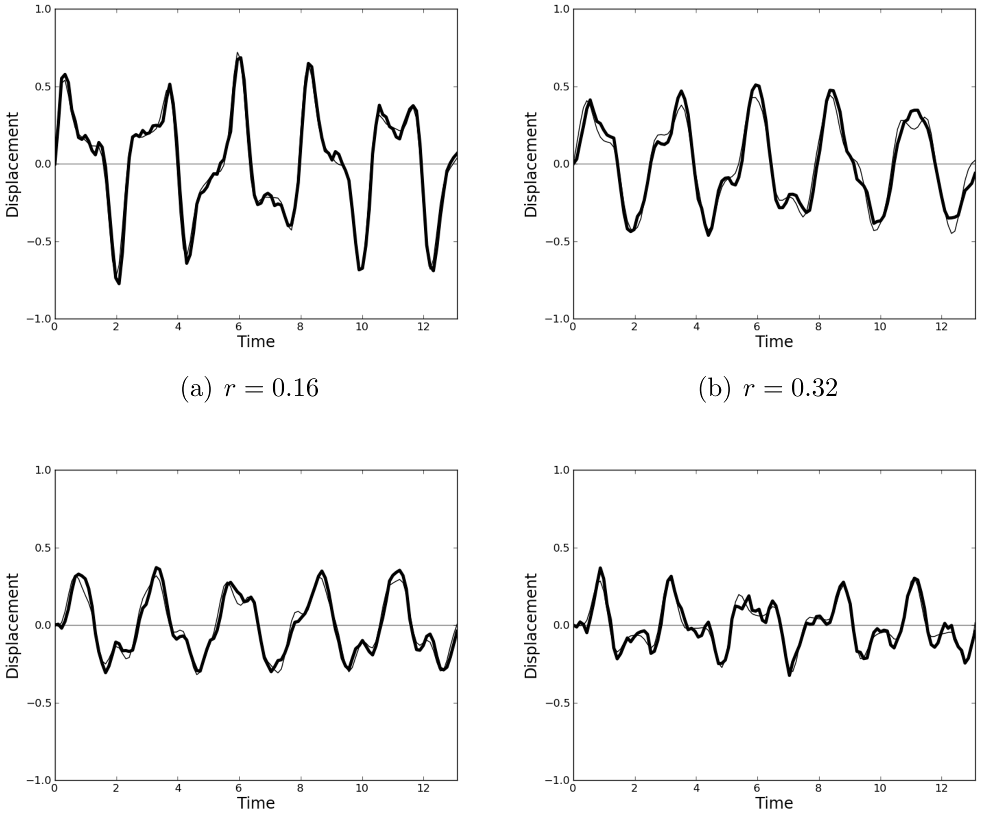
(b) Point (0.3,0.6)

(c) Point (0.6,0.5)

(d) Point (0.9,0.8)

Figure 16. Response histories for the rectangular linear membrane.

The non-uniform impulse excited in the circular membrane case was more challenging for the numerical model. The responses of nodes at radius $r = 0.16, 0.32, 0.54$, and 0.73 are depicted in Figure 17. The thick line depicts the numerical prediction. The thin line depicts the analytical solution including the first seven modes. The axis limits for all of the plots are consistent to illustrate relative magnitudes. High-frequency noise was clearly evident, particularly around sharp slope breaks of the lower modes. Despite the high-frequency oscillations, the lower modes were accurately predicted throughout the run.



(a) $r = 0.16$

(b) $r = 0.32$

(c) $r = 0.54$

(d) $r = 0.73$

Figure 17. Response histories for the circular linear membrane.

Smoothing the Response Histories.

The intensive effort over the decades to optimize integrators for dissipating high frequency content has been absolutely crucial for predictor algorithms, where errors due to noise can accumulate through time steps. In this simultaneous TCG method, however, they did not adversely impact the generally good performance of the model at the targeted frequencies. The spurious modes had a negligible effect on the energy of the system during the short runs of this study.

So what of the requirement that an integrator dissipate the unwanted high-frequency content? A previous TCG study demonstrated that dissipation could be handled outside of the primary integration algorithm. The desired effects were achieved with a dual-algorithm technique — an accurate high-order technique with little to no damping provided the bulk of the response history, while a second dissipative lower-order method applied over several time steps prior to a recorded event stabilized the solution [62]. Depending on the goals of the analysis, the present method can be employed as part of a two-step process: accurate simultaneous TCG solution followed by the application of a low-pass filter. The low-pass filter smooths the response histories by eliminating the undesired high-frequency content.

The options available for a low-pass filter are endless and we demonstrate only one: the Savitzky-Golay filter [99, 114, 104]. Used primarily in spectroscopy, the Savitzky-Golay filter minimizes distortion of the signal while reducing the noise. Most filters operate in the frequency domain; in contrast, the Savitzky-Golay filter operates in the time domain by performing local least-squares polynomial fits throughout the time series. Parameters include the order of the polynomial n and the size of the window N inside which the local regression is performed. The window size, also called the filter length, is defined as number of nodes in the window. The window extends symmetrically $\pm k\Delta t$ from node i where k is an integer; hence $N = 2k + 1$. Given the minimum reliable wavelength T from Eq. 70 and the definition $CFL \equiv \Delta t/(h/c)$, the ideal window size in terms of time increments is the reciprocal of the normalized frequency.

$$\frac{T}{\Delta t} = \frac{4\pi}{3}\left(\frac{1}{CFL}\right) \tag{73}$$

In converting the ideal window size to an odd integer N for use in the algorithm, conservative rounding procedures minimized the loss of desired content. The operator $floor$ returns the largest integer less than the argument, and *odd* returns the largest odd integer less than the argument.

$$N = \text{odd}\left(\text{floor}\left(T/\Delta t\right) - 1\right) \tag{74}$$

For $CFL < CFL_{max}$ this technique ensured a minimum window size of $N = 3$. When $CFL < 0.46$, the large window size resulted in excessively attenuated peaks. Table 3 shows the window sizes as a function of CFL. The filter order was set to one. It is important to note that in general the Savitzky-Golay parameters can be optimized to provide good smoothing of long-wavelength signals or good fidelity of narrow peak shapes, but not both simultaneously [104]. For this demonstration the former goal was chosen.

Table 3. Savitzky-Golay window size determination.

CFL range	Window Size N
$0.46 < CFL < 0.6$	7
$0.6 \leq CFL \leq 0.83$	5
$0.83 < CFL$	3

The effects of the Savitzky-Golay smoothing on the rectangular membrane solution are shown in the frequency domain in Figure 18, and in the time domain in Figure 19 for comparison to the unfiltered responses of Figure 16. Likewise, the smoothed solution for the circular membrane is shown in Figure 20 and in Figure 21, which may be compared to the raw solution of Figure 17. In both cases, the first two modes remained largely intact with only slight attenuation of the peaks. The third and fourth modes were slightly attenuated. For the rectangular membrane, the spurious jumps were removed and the smoothed results very nearly matched the exact solutions. Even for the more challenging circular membrane, the smoothing significantly reduced the noise and the lower modes tracked the exact solution closely.

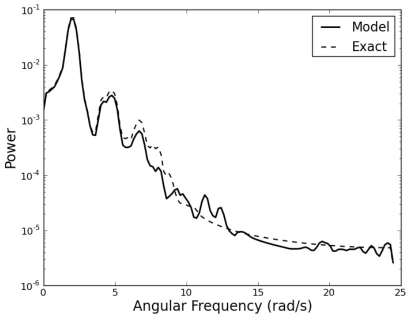
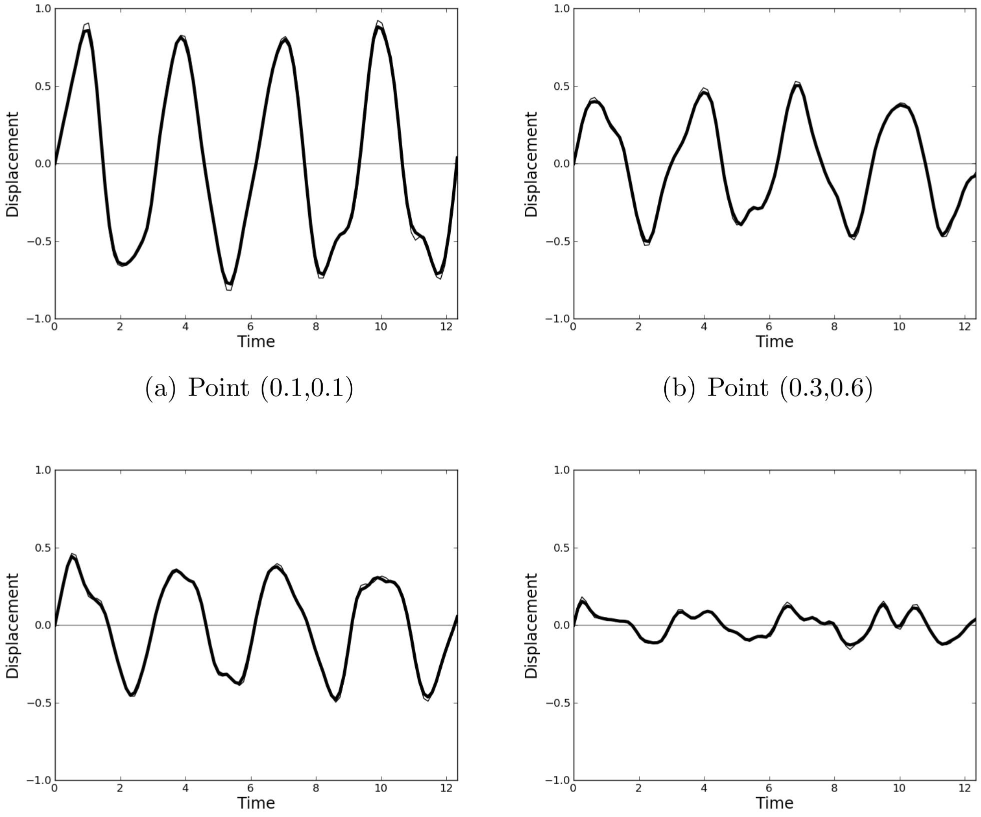


Figure 18. Filtered frequency spectrum for rectangular membrane.

The demonstrated smoothing technique traded some accuracy at the peaks to eliminate most of the noise while preserving the targeted frequency spectra. The selection and design of the filter were guided by the observed limitations of the model. Many alternative filtering techniques are available, but none should be applied arbitrarily. We merely demonstrated that this particular solve-then-smooth process was a viable one for the dynamic cases in this study.

Summary

This study has demonstrated the effectiveness of a baseline STCG formulation in predicting targeted modal frequencies and producing response histories of a linear membrane. A methodology based on signal sampling concepts led to a time increment sufficient for capturing frequency content in a desired range. The time increment was



(a) Point (0.1,0.1)

(b) Point (0.3,0.6)

(c) Point (0.6,0.5)

(d) Point (0.9,0.8)

Figure 19. Filtered response histories for the rectangular linear membrane.

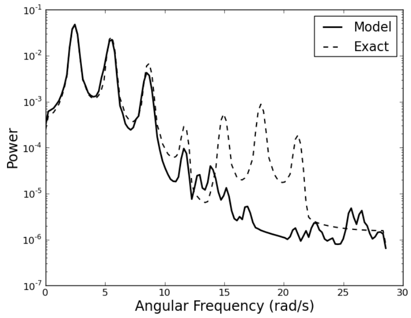
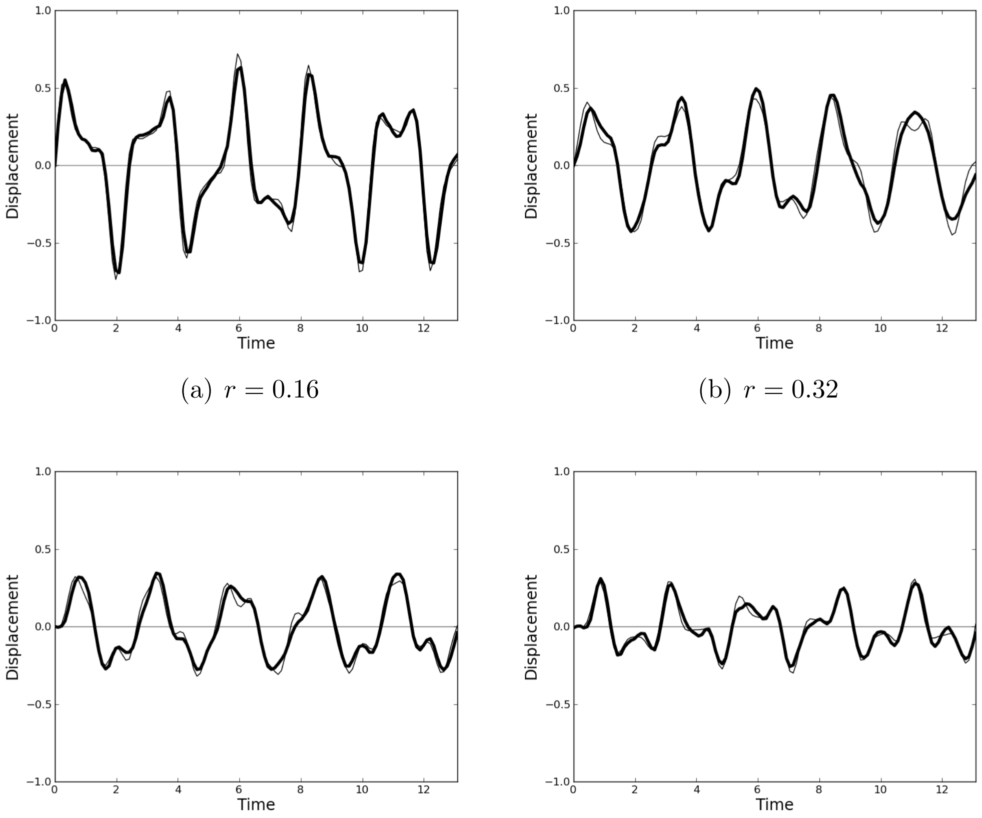


Figure 20. Filtered frequency spectrum for circular membrane.



(a) $r = 0.16$

(b) $r = 0.32$

(c) $r = 0.54$

(d) $r = 0.73$

Figure 21. Filtered response histories for the circular linear membrane.

no more restrictive than that of conventional methods. The spurious high-frequency oscillations were consistent (thus characterizable) and did not cause instability or degrade the lower-frequency accuracy. Post-process filtering was not offered as a panacea or ultimate replacement for an optimal algorithm/element combination. However, for these cases, characterization of the unwanted frequency content led to a smoothing methodology capable of removing much of the noise without affecting the modal frequencies. These results were accomplished with fairly coarse spatial meshes and the simplest of elements.

Although the simultaneous solution approach was analyzed in this work when applied to transient dynamics, there is greater potential for application to periodic problems. Example scenarios include periodic external forces or periodic boundary conditions, either of which could conceivably be induced by a wing flapping mechanism. In this case, the simultaneous solution approach turns an initial value problem (IVP) into a boundary value problem (BVP), enabling the employment of innumerable finite element techniques in the time domain. Based on the results of this study, the method holds promise for such an application.

IV. Nonlinear Analysis of Membrane Dynamics

For numerical simulation of nonlinear membranes, a different approach was taken than the one that led to the Simultaneous Time-Continuous Galerkin (STCG) method of Chapter III. There were several rationales for this decision. First, the nonlinear model was to include both in-plane and out-of-plane displacements, so the size of the global matrices of the STCG would quickly exceed available computer hardware limitations. Second, since the system would no longer be linear, timely convergence to the global solution given an arbitrary initial guess became a concern. Lastly, the high-frequency noise present in the STCG solutions was not catastrophic in the linear case, but could potentially become more troublesome for a nonlinear system.

Instead of the space-time FEA approach, the more common path was taken by separately discretizing space and time to employ a time-marching algorithm. In this way, novel discretization schemes could be devised separately for both space and time, independently studied, and finally combined to fulfill the ultimate objectives of this research. This chapter is organized to follow this logic. First, the Hermite time interpolation scheme is detailed, followed by the point collocation spatial discretization scheme. Finally, the combination of the two schemes to simulate dynamic, geometrically nonlinear membranes is examined.

Hermite Time Interpolation

This section describes a Hermite time interpolation method that differs from previous related works in the literature [109, 81] by its formulation. Rather than using an integral formulation, the proposed method determines the unique quintic polynomial interpolation using beginning and end point constraints on the third derivative of the function, called the "jerk" [123]. This approach opens possibilities for performance optimization by developing appropriate constraint formulas.

Besides proposing an alternative formulation, the previously-mentioned works will be extended by calculating the precise stability boundaries for a linear unforced oscillator, exploring the fundamental source of the approach's instabilities, precisely calculating the dispersion (relative period error), and demonstrating a procedure for accurately estimating local error. To illustrate implementation of the technique, a fixed-point iteration algorithm will be provided as a launching point for general systems of nonlinear partial differential equations. Numerical examples, including multidimensional and stiff systems, will demonstrate the versatility of the method and the utility of the local error estimation procedure.

Time Discretization.

Time is discretized by the dimensionless time τ during the time interval t^0 to t^1. The time step size is $\Delta t = t^1 - t^0$. In this portion of the research, superscripts with a number from 0 to 1 indicate the discrete time at which the value is taken (for example, u^0 is the value of u at the beginning of the time step, where $\tau = 0$).

$$\tau = \frac{t - t^0}{\Delta t} \quad \tau \in [0, 1], t \in \left[t^0, t^1\right] \tag{75}$$

Consider the scalar equation $L(y) = f$, where the differential operator L and forcing function f may be nonlinear, and the solution y is smooth ($y \in C^\infty$). The function value y and its first two derivatives at the beginning and end of the time step are collected into the vector $\mathbf{z}$,

$$\mathbf{z} = \left\{ \begin{array}{cccccc} y(t^0) & \dot{y}(t^0) & \ddot{y}(t^0) & y(t^1) & \dot{y}(t^1) & \ddot{y}(t^1) \end{array} \right\}^T \tag{76}$$

and re-labeled as the discrete nodal variables displacement u, velocity v, and acceleration a.

$$\mathbf{z} = \left\{ \begin{array}{cccccc} u^0 & v^0 & a^0 & u^1 & v^1 & a^1 \end{array} \right\}^T \tag{77}$$

The information in $\mathbf{z}$ is sufficient to determine a quintic polynomial from t^0 to t^1 of the form $u(\tau) = \sum b_i \tau^i$ for $i = 0 \ldots 5$ with constant coefficients $b_i \in \Re$. The resulting six quintic Hermite shape functions $\mathbf{H}(\tau)$ were derived in Ref. [36] and are shown in Eq. 78. For convenience, the first, second, and third derivatives of the Hermite shape functions are provided in Eq. 79, 80, and 81, respectively.

$$\mathbf{H}(\tau) = \left\{ \begin{array}{c} 1 - 10\tau^3 + 15\tau^4 - 6\tau^5 \\ \Delta t \left(\tau - 6\tau^3 + 8\tau^4 - 3\tau^5\right) \\ \Delta t^2 \left(\frac{1}{2}\tau^2 - \frac{3}{2}\tau^3 + \frac{3}{2}\tau^4 - \frac{1}{2}\tau^5\right) \\ 10\tau^3 - 15\tau^4 + 6\tau^5 \\ \Delta t \left(-4\tau^3 + 7\tau^4 - 3\tau^5\right) \\ \Delta t^2 \left(\frac{1}{2}\tau^3 - \tau^4 + \frac{1}{2}\tau^5\right) \end{array} \right\}^T \tag{78}$$

$$\dot{\mathbf{H}}(\tau) = \left\{ \begin{array}{c} \left(-30\tau^2 + 60\tau^3 - 30\tau^4\right)/\Delta t \\ 1 - 18\tau^2 + 32\tau^3 - 15\tau^4 \\ \Delta t \left(\tau - \frac{9}{2}\tau^2 + 6\tau^3 - \frac{5}{2}\tau^4\right) \\ \left(30\tau^2 - 60\tau^3 + 30\tau^4\right)/\Delta t \\ -12\tau^2 + 28\tau^3 - 15\tau^4 \\ \Delta t \left(\frac{3}{2}\tau^2 - 4\tau^3 + \frac{5}{2}\tau^4\right) \end{array} \right\}^T \tag{79}$$

$$\ddot{\mathbf{H}}(\tau) = \begin{Bmatrix} \left(-60\tau + 180\tau^2 - 120\tau^3\right)/\Delta t^2 \\ \left(-36\tau + 96\tau^2 - 60\tau^3\right)/\Delta t \\ 1 - 9\tau + 18\tau^2 - 10\tau^3 \\ \left(60\tau - 180\tau^2 + 120\tau^3\right)/\Delta t^2 \\ \left(-24\tau + 84\tau^2 - 60\tau^3\right)/\Delta t \\ 3\tau - 12\tau^2 + 10\tau^3 \end{Bmatrix}^T \tag{80}$$

$$\dddot{\mathbf{H}}(\tau) = \begin{Bmatrix} \left(-60 + 360\tau - 360\tau^2\right)/\Delta t^3 \\ \left(-36 + 192\tau - 180\tau^2\right)/\Delta t^2 \\ \left(-9 + 36\tau - 30\tau^2\right)/\Delta t \\ \left(60 - 360\tau + 360\tau^2\right)/\Delta t^3 \\ \left(-24 + 168\tau - 180\tau^2\right)/\Delta t^2 \\ \left(3 - 24\tau + 30\tau^2\right)/\Delta t \end{Bmatrix}^T \tag{81}$$

The nodal approximation for the third derivative of y is also given a label (j for "jerk"), so the discrete approximations of the function y and its first three derivatives throughout a time increment $t \in [t_0, t_1]$ are then

$$y(t) \approx u(\tau) = \mathbf{H}(\tau)\mathbf{z} \tag{82}$$

$$\dot{y}(t) \approx v(\tau) = \dot{\mathbf{H}}(\tau)\mathbf{z} \tag{83}$$

$$\ddot{y}(t) \approx a(\tau) = \ddot{\mathbf{H}}(\tau)\mathbf{z} \tag{84}$$

$$\dddot{y}(t) \approx j(\tau) = \dddot{\mathbf{H}}(\tau)\mathbf{z} \tag{85}$$

These interpolation formulas define a trajectory with a continuous acceleration profile. The displacement, velocity, and acceleration curves are intrinsically constrained to each other and have been “pre-integrated” – once the acceleration curve is uniquely determined, the velocity and displacement profiles automatically follow. Also, as will later be demonstrated, the polynomials will supply a wealth of information about the dynamics occurring within the time step that can be exploited for error estimation.

Constraint Formulation.

The use of jerk constraints to define the unique solution forms the foundation of the present method. The user-defined formulas for jerk at times t^0 and t^1 are tied to the Hermite polynomial interpolation, thereby defining the beginning and end slopes of the cubic acceleration polynomial. Thus a unique set of interpolation polynomials for displacement, velocity, and acceleration are defined by the Hermite constraint equations

$$\begin{aligned} \dddot{\mathbf{H}}(0)\mathbf{z} - j^0 = 0 \\ \dddot{\mathbf{H}}(1)\mathbf{z} - j^1 = 0 \end{aligned} \tag{86}$$

It is convenient at this point to remove the acceleration variable from $\mathbf{z}$, as in practice a will be calculated in tandem with j. The resulting matrix form of the Hermite constraint equations is

$$\begin{aligned} &\begin{bmatrix} \frac{60}{\Delta t^3} & -\frac{24}{\Delta t^2} \\ \frac{60}{\Delta t^3} & -\frac{36}{\Delta t^2} \end{bmatrix} \begin{Bmatrix} u^1 \\ v^1 \end{Bmatrix} + \begin{bmatrix} \frac{3}{\Delta t} & 0 \\ \frac{9}{\Delta t} & -1 \end{bmatrix} \begin{Bmatrix} a^1 \\ j^1 \end{Bmatrix} \\ = &\begin{bmatrix} \frac{60}{\Delta t^3} & \frac{36}{\Delta t^2} \\ \frac{60}{\Delta t^3} & \frac{24}{\Delta t^2} \end{bmatrix} \begin{Bmatrix} u^0 \\ v^0 \end{Bmatrix} + \begin{bmatrix} \frac{9}{\Delta t} & 1 \\ \frac{3}{\Delta t} & 0 \end{bmatrix} \begin{Bmatrix} a^0 \\ j^0 \end{Bmatrix} \end{aligned} \tag{87}$$

For some problems the equation can be simplified by eliminating the a and j terms by substitution and solving the linear system directly. Otherwise, moving the a^1 and j^1 terms to the right-hand side provides an expression for solving the system by fixed-point iteration. With no effect on previous developments, the scalar degrees of freedom are now shown as vectors to generalize the scheme for multidimensional problems.

$$\begin{Bmatrix} \mathbf{u}^1 \\ \mathbf{v}^1 \end{Bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \mathbf{u}^0 \\ \mathbf{v}^0 \end{Bmatrix} + \begin{bmatrix} 7\Delta t^2/20 & \Delta t^3/20 \\ \Delta t/2 & \Delta t^2/12 \end{bmatrix} \begin{Bmatrix} \mathbf{a}^0 \\ \mathbf{j}^0 \end{Bmatrix} + \begin{bmatrix} 3\Delta t^2/20 & -\Delta t^3/30 \\ \Delta t/2 & -\Delta t^2/12 \end{bmatrix} \begin{Bmatrix} \mathbf{a}^1 \\ \mathbf{j}^1 \end{Bmatrix} \tag{88}$$

The fixed-point method shown above can also accurately be described as successive substitution between two sets of equations [50], as shown clearly by the following algorithm:

1. From $\mathbf{u}^0$ and $\mathbf{v}^0$, calculate $\mathbf{a}^0$ and $\mathbf{j}^0$ (unless carried over from previous time step)
2. Guess $\mathbf{u}^1$ and $\mathbf{v}^1$
3. Calculate $\mathbf{a}^1$ and $\mathbf{j}^1$ (discussed in the next section)
4. Until converged:
    (a) Calculate $\mathbf{u}^1$ an $\mathbf{v}^1$ using Eq. (88)
    (b) If converged, STOP; else, update $\mathbf{a}^1$ and $\mathbf{j}^1$

The fixed point iteration method only requires the calculation of explicit vector formulas – the tasks of managing a Jacobian and solving a linear system at each iteration is avoided. Convergence will be slower than the Newton method's quadratic rate, but acceleration methods are available if required [135]. Newton iterations are more robust and may converge in cases where fixed point iteration fails. Generally, the efficacy of fixed-point iteration versus Newton iterations tends to be problem-specific and comparisons may be found in the literature [116, 48].

Keeping with the single-step perspective, the initial guesses for $\mathbf{u}^1$ and $\mathbf{v}^1$ in this study were formed from the available information at time t^0.

$$\mathbf{u}^1 = \mathbf{u}^0 + \mathbf{v}^0 \Delta t + \tfrac{1}{2}\mathbf{a}^0 \Delta t^2 + \tfrac{1}{6}\mathbf{j}^0 \Delta t^3 \tag{89}$$

$$\mathbf{v}^1 = \mathbf{v}^0 + \mathbf{a}^0 \Delta t + \tfrac{1}{2}\mathbf{j}^0 \Delta t^2 \tag{90}$$

The stopping criterion was based on a tolerance ϵ for the change in the solution vector after an update. Letting Δ indicate the change in a value after an iteration, iterations were stopped when

$$\max\left(\left|\Delta\mathbf{u}\right|_{max}, \left|\Delta\mathbf{v}\right|_{max}\right) < \epsilon \tag{91}$$

Since accuracy was the focus of the study, tight tolerances were generally set without consideration of the number of function calls required.

Constraint Definition.

Given the Hermite interpolation scheme just described, constraints will be applied at the end points to obtain a unique solution. Consider a single time step where the initial displacement u^0 is known, and the initial velocity v^0 is given or may be calculated from the first-order ODE. Four unknowns remain in the vector $\mathbf{z}$: a^0, u^1, v^1, and a^1. Two equations may be provided by constraining the accelerations in accordance with the ODE or knowledge of the particular physical system.

$$\begin{aligned} a^0 &= \ddot{y}\left(t^0, u^0, v^0, f^0\right) \\ a^1 &= \ddot{y}\left(t^1, u^1, v^1, f^1\right) \end{aligned} \tag{92}$$

When posed as a nonlinear structural engineering problem with constant mass matrix $\mathbf{M}$, damping matrix $\mathbf{C}$, stiffness matrix $\mathbf{K}$, and external load $\mathbf{F}$ the equilibrium equation is

$$\mathbf{Ma} + \mathbf{Cv} + \mathbf{Ku} + \mathbf{F} = 0 \tag{93}$$

The familiar acceleration constraint formulas are obtained by rearranging the equilibrium equation.

$$\begin{aligned} \mathbf{a^0} &= -\mathbf{M^{-1}} \left(\mathbf{C^0 v^0} + \mathbf{K^0 u^0} + \mathbf{F^0} \right) \\ \mathbf{a^1} &= -\mathbf{M^{-1}} \left(\mathbf{C^1 v^1} + \mathbf{K^1 u^1} + \mathbf{F^1} \right) \end{aligned} \tag{94}$$

Now only u^1 and v^1 are unknown. To obtain the final two equations, j^0 and j^1 must be defined. There is great freedom in choosing jerk constraints, as long as desired performance is achieved. To illustrate this point, suppose requirements called for an algorithm capable of unconditional stability and controllable dissipation. The derivation could lead to the jerk constraints

$$j^0 = \left[60 \left(\beta - \frac{3}{20} \right) - 24 \left(\gamma - \frac{1}{2} \right) \right] \left(\frac{a^1 - a^0}{\Delta t} \right) \tag{95}$$

$$j^1 = \left[60 \left(\beta - \frac{3}{20} \right) - 36 \left(\gamma - \frac{1}{2} \right) \right] \left(\frac{a^1 - a^0}{\Delta t} \right) \tag{96}$$

where γ and β are scalar parameters. These jerk constraints revert the present formulation to the classic Newmark formulation. (This correspondence can easily be shown by side-by-side comparison of the Newmark equations with Eq. 88.) For non-dissipative Newmark methods, where $\gamma = 1/2$, the constraints are identical and an elegant symmetry is obtained.

$$j^0 = j^1 = 60 \left(\beta - \frac{3}{20} \right) \left(\frac{a^1 - a^0}{\Delta t} \right) \tag{97}$$

The Newmark jerk constraints obviously provide excellent results, but they result in artificially oscillatory Hermite interpolations within the time step. Greater accuracy can be achieved by defining jerk according to its physical and mathematical definition: as the time derivative of the acceleration constraints in Eq. (92).

$$\begin{aligned} j^0 &= \dddot{y}\left(t^0, u^0, v^0, a^0, f^0, \dot{f}^0\right) \\ j^1 &= \dddot{y}\left(t^1, u^1, v^1, a^1, f^1\dot{f}^1\right) \end{aligned} \tag{98}$$

The jerk constraints are stated here in terms of acceleration, which may be eliminated by substitution of Eqs.(94) if desired.

$$\begin{aligned} \mathbf{j}^0 &= -\mathbf{M}^{-1}\left[\mathbf{C}^0\mathbf{a}^0 + \left(\dot{\mathbf{C}}^0 + \mathbf{K}^0\right)\mathbf{v}^0 + \dot{\mathbf{K}}^0\mathbf{u}^0 + \dot{\mathbf{F}}^0\right] \\ \mathbf{j}^1 &= -\mathbf{M}^{-1}\left[\mathbf{C}^1\mathbf{a}^1 + \left(\dot{\mathbf{C}}^1 + \mathbf{K}^1\right)\mathbf{v}^1 + \dot{\mathbf{K}}^1\mathbf{u}^1 + \dot{\mathbf{F}}^1\right] \end{aligned} \tag{99}$$

For the remainder of this dissertation, although the Hermite interpolation method has been shown to be a family of methods depending on the chosen jerk constraints, the method will be confined to the jerk constraints of Eqs. (98).

Stability and Energy Conservation.

The linear, undamped, free system $\ddot{y} + \omega^2 y = 0$ with constant angular frequency ω provides the prototypical case for investigating the method's stability and energy conservation. The constraint equations trivially fall from the ODE as $a = -\omega^2 u$ and $j = -\omega^2 v$. The constraint equations are substituted into Eq. 88 to produce the iteration matrix $\mathbf{A}$, which for a time step maps the initial state $\{\mathbf{u}^0, \mathbf{v}^0\}^T$ to the final state $\{\mathbf{u}^1, \mathbf{v}^1\}^T$. The time step size is non-dimensionalized as $\nu = \omega\Delta t = 2\pi\Delta t/T$, where T is the period of the oscillator. While ν simplifies the matrices, most of the following discussion will reference $\Delta t/T$, due to its widespread use in the finite element literature. The iteration matrix simplifies to

$$\mathbf{A} = \frac{1}{\nu^4 + 16\nu^2 + 240} \begin{bmatrix} 3\nu^4 - 104\nu^2 + 240 & \left(\frac{1}{\omega}\right)\left(\frac{1}{3}\nu^5 - 24\nu^3 + 240\nu\right) \\ -\omega\left(-24\nu^3 + 240\nu\right) & 3\nu^4 - 104\nu^2 + 240 \end{bmatrix} \tag{100}$$

which is the method's approximation of the exact iteration matrix

$$\mathbf{A}_{exact} = \begin{bmatrix} \cos\left(\nu\right) & \left(\frac{1}{\omega}\right)\sin\left(\nu\right) \\ -\omega\sin\left(\nu\right) & \cos\left(\nu\right) \end{bmatrix} \tag{101}$$

The exact iteration matrix is typically found in Hamiltonian formulations using the conjugate momentum p, which is equivalent to this formulation by assuming without loss of generality that the mass $m = 1$, so $v = p$. [116, 48]. For this unforced linear oscillator, the present method produces an iteration matrix with the same diagonal elements as the weighted residual method of Razavi *et al.* [109]. Therefore, the stability characteristics will be the same and their findings can be further developed. An in-depth examination of the iteration matrix follows, to provide greater insight into the method.

The components of the exact iteration matrix and its approximation are compared in Figure 22. The horizontal axes represent a chosen step size $\Delta t/T$, and the vertical axis is the value of the corresponding matrix element. The dashed lines are used for the exact matrix, $\mathbf{A}_{exact}$, and the solid lines are used for the approximate matrix, $\mathbf{A}$. The plus sign markers show the values of the iteration matrix obtained from Ref. [109]. The absolute value of the errors is shown in the bottom right subplot up to the Nyquist frequency (note the change to a logarithmic vertical axis). The approximations are quite accurate for reasonable time steps, with error magnitudes of the matrix elements less than 1×10^{-4} for $\Delta t/T = 0.1$. However, the downside of using this polynomial approximation is also obvious, as with increasing time step size, the approximations exceed the bounding range of the exact values and $\mathbf{A}_{12}$ grows linearly without bound. The regions of instability defined by this behavior will be characterized next.

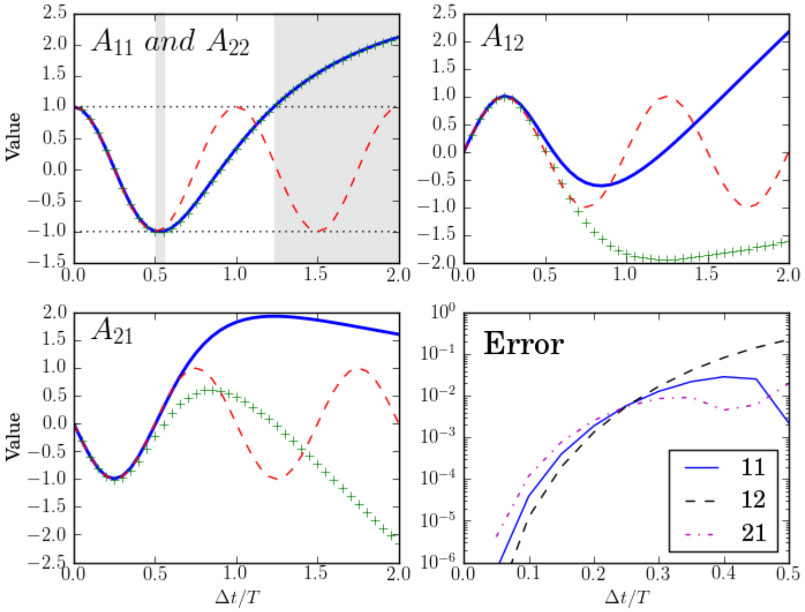


Figure 22. Graphical depiction of the Hermite time interpolation method iteration matrix for the simple harmonic oscillator

A symplectic method accurately reproduces the geometric structure of the ODE and its solutions, and conserves invariants over long-duration simulations [48]. For Hamiltonian systems in mechanics the invariant is typically energy. It is easily proven that the iteration matrix $\mathbf{A}$ is symplectic because for all ω and ν it satisfies the relationship

$$\mathbf{A}^T\mathbf{J}\mathbf{A} = \mathbf{J} \tag{102}$$

where $\mathbf{J}$, the skew-symmetric operator used when representing a Hamiltonian system of equations in matrix form, is defined as

$$\mathbf{J} \equiv \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \tag{103}$$

As with any symplectic matrix, it is also true that $det(\mathbf{A}) = 1$ [89]. Since A is an area-preserving linear mapping, the stability can be determined by finding where $|tr(\mathbf{A})| < 2$ [2]; or, since $A_{11} = A_{22}$, where $|A_{11}| < 1$. Hence the stability can be inspected in the upper-left plot of Figure 22. It is immediately observed that there are two regions of instability – a small region just above $\Delta t/T = 0.5$, and a clear limit above $\Delta t/T \approx 1.2$, corresponding to those of Ref. [109]. The regions of instability are shaded gray in the figure. We now determine the exact locations of the stability limits. The boundaries are determined from the polynomial inequality

$$\left| \frac{3\nu^4 - 104\nu^2 + 240}{\nu^4 + 16\nu^2 + 240} \right| < 1 \tag{104}$$

The terms with ν can be lumped into a single term,

$$\left| 1 + \frac{2\nu^4 - 120\nu^2}{\nu^4 + 16\nu^2 + 240} \right| < 1 \tag{105}$$

with some final manipulations showing that the stability requires satisfaction of two inequalities.

$$\frac{2\nu^4 - 120\nu^2}{\nu^4 + 16\nu^2 + 240} > -2 \tag{106}$$

$$\frac{\nu^4 - 60\nu^2}{\nu^4 + 16\nu^2 + 240} < 0 \tag{107}$$

The first statement is violated if $10 < \nu^2 < 12$ and the second is violated if $\nu^2 > 60$, providing the exact instability regions in terms of sampling frequency as

$$\sqrt{5/(2\pi^2)} < \Delta t/T < \sqrt{3/\pi^2} \tag{108}$$

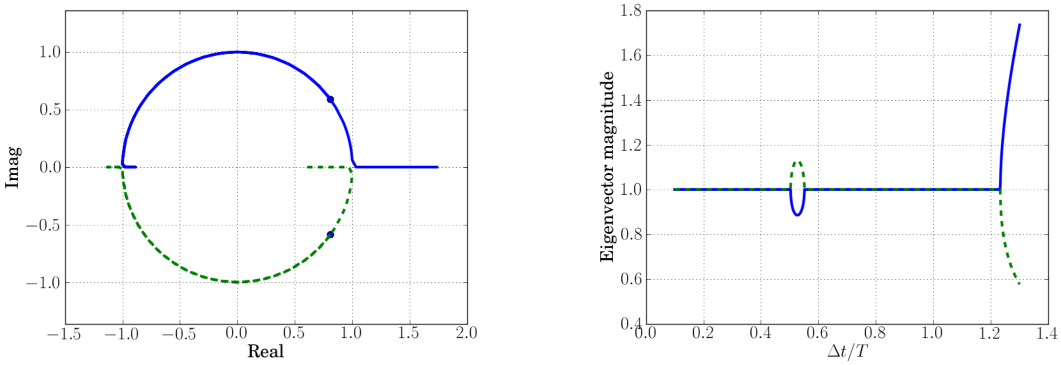
$$\sqrt{15/\pi^2} < \Delta t/T \tag{109}$$

The instability can be further characterized by an eigenvalue analysis. The expression for the eigenvalues is

$$\lambda_i = \frac{3\nu^4 - 104\nu^2 + 240 \pm 2\nu\sqrt{2\left(\nu^2-60\right)\left(\nu^2-12\right)\left(\nu^2-10\right)}}{\nu^4 + 16\nu^2 + 240} \tag{110}$$

and the stability boundaries can be seen by inspection of the expression under the radical. When the radicand is positive the eigenvalues lie on the real axis of the complex plane and instability results [28]. The path of the eigenvalues as the time step is increased is depicted in Figure 23(a). Starting at the dots on the right-hand side and increasing the step size, the eigenvalues follow the unit circle in the negative-real direction until they collide and separate at $(-1, 0)$, corresponding to $\Delta t/T = \sqrt{5/(2\pi^2)}$. They quickly return to the circle and follow it to $(1, 0)$, at which point they again collide and repel each other. They remain on the real axis as $\Delta t/T \to \infty$. For a detailed discussion of this behavior, see Ref. [2].

The moduli of the two eigenvalues as a function of $\Delta t/T$ are plotted in Figure 23(b). The spectral radius $\rho(\mathbf{A})$, defined as the maximum of the moduli, is equal to one wherever the method is stable – no numerical damping is present and the energy is conserved. The traces also show that after bifurcation, the moduli are the reciprocals of each other, another indicator of a symplectic transformation [2]. Exploring the nature of the instability provides a launching point for devising mitigation techniques. A small amount of physical damping can eliminate the small unstable region just above the Nyquist frequency [109]. However, physical damping alone af-



(a) Root locus trace of the two eigenvalues.

(b) Spectral radius of the iteration matrix **A**.

Figure 23. Eigenvalue behavior for the simple harmonic oscillator.

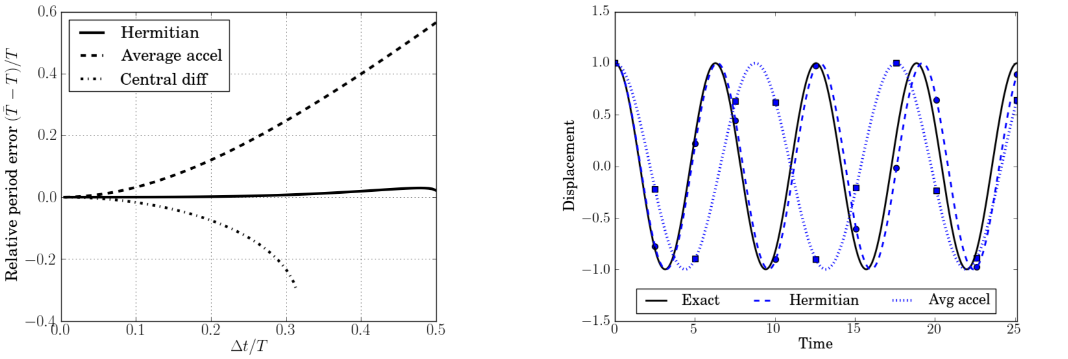
fects only middle frequencies and cannot remove high frequency instabilities [58] – algorithmic damping would be required, a topic left for future research.

Dispersion.

To quantify the method's dispersion, let $\overline{T}$ be the period of the model solution and T be the exact period. Then the period error is $P = \overline{T}/T$ [28] and the relative period error is $(\overline{T} - T)/T$ [58]. The period error P can be calculated from the eigenvalues of Eq. 110 as

$$P = \frac{\overline{T}}{T} = \nu \left[\arctan \left(\frac{2\nu\sqrt{2\left(\nu^2 - 60\right)\left(\nu^2 - 12\right)\left(\nu^2 - 10\right)}}{3\nu^4 - 104\nu^2 + 240} \right) \right]^{-1} \tag{111}$$

As shown in Figure 24(a), the Hermite method accurately reproduces the frequency of the oscillator up to the Nyquist frequency with a maximum relative period error of 0.030. The error is less than 0.0016 below $\Delta t/T = 0.2$. The central difference shrinks the period until it hits its stability limit at $\Delta t/T = 1/\pi$ [28]. The average acceleration method significantly increases the period. To illustrate the extent, Figure 24(b) portrays the period errors for a simple harmonic oscillator using the Hermite and average acceleration methods at $\Delta t/T = 0.4$. Because the numerical samples are so sparse, least-square fits of sinusoidal functions to each method are included in the figure. The fidelity of frequency reproduction bodes well for the application of this method in a wave propagation problem.



(a) Relative period errors.

(b) Example of period error at $\Delta t/T = 0.4$.

Figure 24. Relative period error for the simple harmonic oscillator comparing various integration methods.

Local Error Estimation.

Error estimates can alert the user to potentially inaccurate solutions, trigger damping mechanisms, or direct step size changes. A common technique is to compare solutions from two numerical methods of different order [3]. The Runge-Kutta 4/5 has proven useful because it does so with only one more function call [117]. Since the present method enforces constraints at the end points of a time step to dictate the solution, the degree to which those constraints are violated in the interior offers useful data for error estimation. An example will be developed here which uses one additional function call at mid-step to estimate the local error.

Suppose the solution for a time step has been obtained so the vector $\mathbf{z}$ is known. Let the interpolated value of acceleration at the midpoint $\tau = 0.5$ be $a^{0.5} = \ddot{\mathbf{H}}(0.5)\mathbf{z}$. Also at the midpoint, the acceleration constraint equation is applied to compute the constraint value of acceleration $\breve{a}^{0.5} = \ddot{y}\left(t^{0.5}, u^{0.5}, v^{0.5}, f^{0.5}\right)$. Thus the acceleration error δa is the difference between the constrained acceleration and the interpolated acceleration at the midpoint of the time step.

$$\delta a = a^{0.5} - \breve{a}^{0.5} = \ddot{\mathbf{H}}(0.5)\mathbf{z} - \ddot{y}\left(t^{0.5}, u^{0.5}, v^{0.5}, f^{0.5}\right) \tag{112}$$

For a consistent formulation, as the time step approaches zero, the interpolation polynomials approach the exact solution throughout the time step and δa approaches zero. The acceleration error will now be used to build an alternative acceleration profile which will be integrated twice to produce a useful error parameter in terms of displacement. First, three-point Lobatto integration puts the acceleration error in terms of velocity. The alternative acceleration polynomial is assumed to pass through the the points (t^0, a^0), $(t^{0.5}, \breve{a}^{0.5})$, and $(t^1, \breve{a}^1)$ where $\breve{a}^1 = a^1 + \delta a$. The Lobatto integration formula then provides a new estimate for the final velocity, v_L^1.

$$v_L^1 = v^0 + \frac{\Delta t}{2}\left(\tfrac{1}{3}a^0 + \tfrac{4}{3}\breve{a}^{0.5} + \tfrac{1}{3}\breve{a}^1\right) \tag{113}$$

Finally, the local error estimate for displacement, δu, is calculated by a simple approximation of the integral of the velocity error through the time step.

$$\delta u = \left|\frac{\Delta t}{2}\left(v^1 - v_L^1\right)\right| \tag{114}$$

To correlate the local error estimate with the exact local error, numerical experiments were conducted using a simple harmonic oscillator. For the experiment, 1,500 data points were created by sampling each of the time step sizes $\Delta t/T = 0.05$, 0.1, 0.2, 0.3, and 0.4 with 300 samples for the phase $\phi = [0, 2\pi]$ where $y = cos(t - \phi)$. A scatter plot of the relationship is shown in Figure 25. Nearly all of the 1,500 data points lie near the line of slope one. As the error estimate δu increases past 0.001, the error is

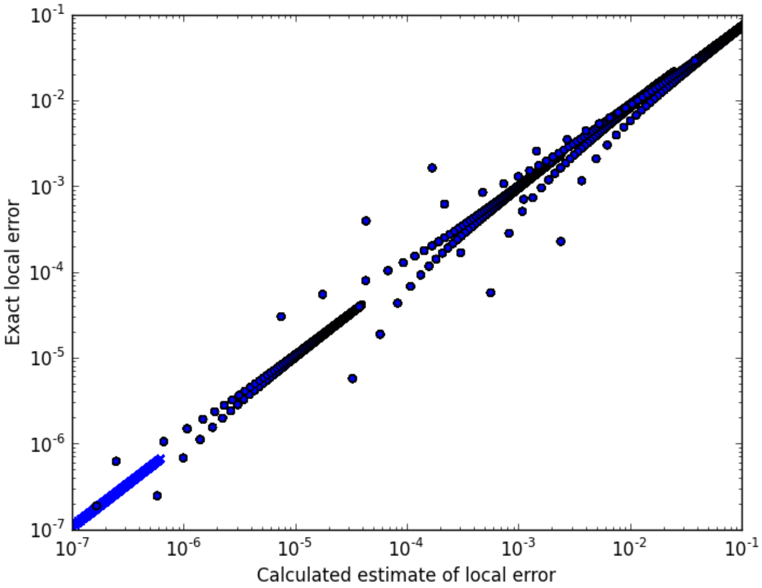


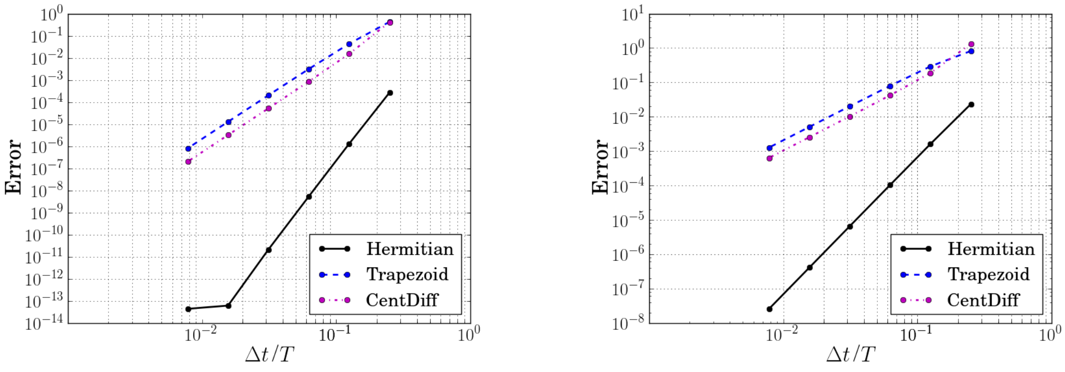
Figure 25. Relationship between the calculated local error estimate and the exact local error for the simple harmonic oscillator.

increasingly overestimated. The sets of points for each $\Delta t/T$ have a forked appearance from the process of converting elliptical plots of the estimated versus exact error to the log-log plot shown here. The error estimation process just described was used for the numerical examples of this paper because of the strong correlation between the estimated and exact local error for reasonable time step sizes and the conservative overestimation of error at extreme step sizes.

Numerical Examples.

Simple Harmonic Oscillator.

The simple harmonic oscillator discussed earlier was solved from two starting conditions: $(y(0), \dot{y}(0)) = (1, 0)$ and $(0, 1)$. For comparison, solutions were also obtained using the two members of the Newmark family: the central difference method, and the trapezoid method with $\gamma = 1/2$ and $\beta = 1/4$ [102]. Convergence for all three methods is shown in Figure 26, and the Hermite interpolation accuracy is clearly superior to the second-order Newmark methods. The observed rate of convergence matched the rates observed using other Hermite polynomial methods [109, 81].



(a) $y(0) = 1$, $\dot{y}(0) = 0$

(b) $y(0) = 0$, $\dot{y}(0) = 1$

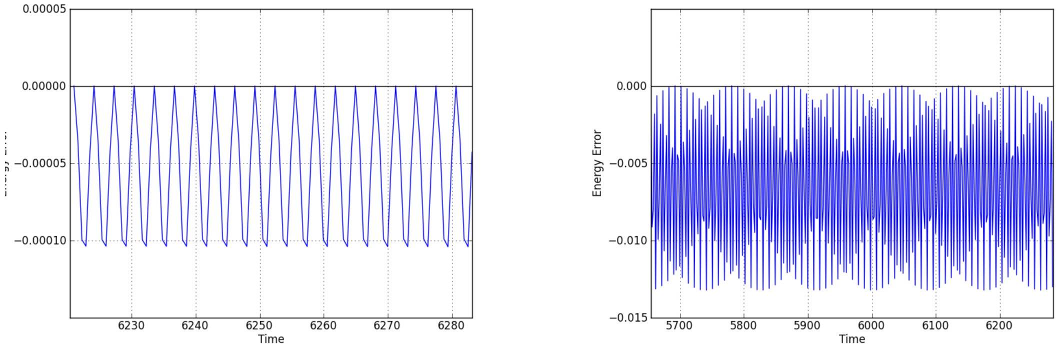
Figure 26. Convergence for the simple harmonic oscillator from two different initial conditions.

As expected after finding the iteration matrix to be symplectic by way of Eq. (102), the energy of the oscillator was conserved over long durations. Figure 27 shows the energy error at the end of 1000 periods, with the left plot depicting the last ten periods with $\Delta t/T = 0.1$, and the right plot depicting the last 100 periods with $\Delta t/T = 0.3$. The error oscillates but remains bounded in a tight range, typical behavior for symplectic methods [68].

Nonlinear Second-Order ODE.

In Ref. [3], the rates of convergence of several techniques were compared by solving a second-order nonlinear ODE. The present method will now be applied to the same problem with two objectives: (1) confirm the order of convergence for the method, and (2) demonstrate its application in a fixed-point iterative algorithm. The ODE is

$$\ddot{y} = -e^{y+1} \tag{115}$$



(a) $\Delta t/T = 0.1$

(b) $\Delta t/T = 0.3$

Figure 27. Energy errors at the conclusion of a 1000-period simulation for two different time step sizes.

with the initial conditions $y(0) = 0$ and $\dot{y}(0) = \theta \tanh(\theta/4)$. By setting the parameter $\theta = 3.03623184819656$, the exact solution for the convergence study is $y(1) = 0$. The ODE directly provides the acceleration constraints of Eq. (92). The time derivative of the ODE provides the jerk constraints of Eq. (98).

$$a \quad = -e^{u+1} \tag{116}$$

$$j \quad = -ve^{u+1} \tag{117}$$

These constraint formulas contribute the acceleration and jerk values to the right-hand side of Eq. 88. The resulting matrix equation, shown in expanded form in Eq. 118, was used in the fixed-point iteration algorithm to obtain the solutions. The subscript k indicates current values, while $k+1$ indicates the new values upon completion of the iteration. The errors and rates of convergence in Table 4 confirm the fourth-order convergence seen in the simple harmonic oscillator. In comparison to the methods presented in Ref. [3], the error magnitudes were approximately the same as those of the fourth-order Runge-Kutta method, but not quite as good as the four-step, fifth-order Adams-Moulton method.

$$\begin{bmatrix} \frac{60}{\Delta t^3} & -\frac{24}{\Delta t^2} \\ \frac{60}{\Delta t^3} & -\frac{36}{\Delta t^2} \end{bmatrix} \begin{Bmatrix} u^1 \\ v^1 \end{Bmatrix}_{k+1} = \begin{bmatrix} \frac{60}{\Delta t^3} & \frac{36}{\Delta t^2} \\ \frac{60}{\Delta t^3} & \frac{24}{\Delta t^2} \end{bmatrix} \begin{Bmatrix} u^0 \\ v^0 \end{Bmatrix} + \begin{bmatrix} \frac{9}{\Delta t} & 1 \\ \frac{3}{\Delta t} & 0 \end{bmatrix} \begin{Bmatrix} -e^{u^0+1} \\ -v^0 e^{u^0+1} \end{Bmatrix} - \begin{bmatrix} \frac{3}{\Delta t} & 0 \\ \frac{9}{\Delta t} & -1 \end{bmatrix} \begin{Bmatrix} -e^{u^1+1} \\ -v^1 e^{u^1+1} \end{Bmatrix}_k \tag{118}$$

Table 4. Convergence results for the exponential function.

$\log(\Delta t)$	log(error)	p
0	-1.01	
-1	-5.28	4.27
-2	-9.27	3.99
-3	-13.26	3.99

System of Nonlinear Equations.

The next example was chosen to demonstrate the present method's applicability to systems of equations and compare the error estimation accuracy to other techniques. This particular problem was taken from Reference [77], where it is used to detail accuracy and error estimation for several integration techniques. For direct comparison to the author's presentation, the local error was calculated as follows. For each time step (not just the first one), the initial values u_i^0 and v_i^0 were set to the exact solution. Thus, at the conclusion of the time step, the exact local error was precisely the difference between the model's solution (u_i^1 and v_i^1) and the exact solution ($y_i(t^1)$ and $\dot{y}_i(t^1)$). The exact local error can then be compared to the estimate provided by the model. The stated error is the L_2 norm of the displacement error vector. The system consists of two nonlinear equations, with $t \in [0, 6.4]$:

$$\ddot{y}_1 = -y_1 \left(y_1^2 + y_2^2\right)^{-3/2} \tag{119}$$

$$\ddot{y}_2 = -y_2 \left(y_1^2 + y_2^2\right)^{3/2} \tag{120}$$

The initial conditions $y_1(0) = 1$, $\dot{y}_1(0) = 0$, $y_2(0) = 0$, and $\dot{y}_2(0) = 1$ provide the exact solution $y_1 = \cos(t)$ and $y_2 = \sin(t)$. As with the previous example, the ODE directly provides the acceleration constraints and the time derivative of the ODE provides the jerk constraints to be applied at the beginning and end of the time steps (Eq. (92) and Eq. (98), respectively). The superscript time indices are omitted for clarity.

$$a_1 = -u_1 \left(u_1^2 + u_2^2\right)^{-3/2} \tag{121}$$

$$a_2 = -u_2 \left(u_1^2 + u_2^2\right)^{3/2} \tag{122}$$

$$j_1 = -v_1 \left(u_1^2 + u_2^2\right)^{-3/2} + 5u_1 \left(u_1 v_1 + u_2 v_2\right) \left(u_1^2 + u_2^2\right)^{-5/2} \tag{123}$$

$$j_2 = -v_2 \left(u_1^2 + u_2^2\right)^{3/2} - u_2 \left(u_1 v_1 + u_2 v_2\right) \left(u_1^2 + u_2^2\right)^{1/2} \tag{124}$$

The system was solved using the fixed-point iteration scheme (Eq. 88) with an iteration stopping tolerance of 1×10^{-10}. The results are shown in Table 5. The present method was more accurate than the reference's examples by an order of magnitude or more. The local error estimates underestimated the exact local error by an average magnitude of 7%, 5%, and 6% for $\Delta t = 0.8$, 0.4, and 0.2 respectively. By comparison, for $\Delta t = 0.2$, the Fehlberg (4,5) Runge-Kutta method average local error was 3.7×10^{-6} and the error estimate was off by an average of 18% [77].

To check convergence rates of the method once again, the simulation was next run without performing the exact solution reset used above for local error estimation analysis. The errors were thus the difference between the model results and the

Table 5. Local error E and local error estimates E_{est} for the system of ODEs.

	$\Delta t = 0.8$		$\Delta t = 0.4$		$\Delta t = 0.2$	
t	E $\times 10^4$	E_{est} $\times 10^4$	E $\times 10^6$	E_{est} $\times 10^6$	E $\times 10^8$	E_{est} $\times 10^8$
0.8	1.83	1.65	2.79	2.61	4.41	4.14
1.6	1.44	1.52	2.64	2.57	4.36	4.13
2.4	1.48	1.53	2.72	2.59	4.40	4.14
3.2	1.83	1.63	2.85	2.63	4.45	4.15
4.0	1.81	1.64	2.77	2.61	4.41	4.14
4.8	1.42	1.51	2.64	2.57	4.36	4.13
5.6	1.50	1.53	2.73	2.59	4.40	4.14
6.4	1.85	1.64	2.86	2.63	4.45	4.15

exact solution at the final time, $t = 6.4$. Convergence rate calculations are shown in Table 6. The observed rates of convergence approached four as the time steps decreased, confirming the findings from the previous examples.

Table 6. Convergence results for the multidimensional example. The order of convergence is labeled p.

Δt	$\log(\Delta t)$	$\log(y_1$ error)	p	$\log(y_2$ error)	p	$\log(L_2$ error)	p
0.8	-0.097	-2.40	–	-2.76	–	-2.36	–
0.4	-0.398	-3.54	3.76	-3.86	3.67	-3.49	3.74
0.2	-0.699	-4.72	3.95	-5.04	3.92	-4.68	3.94

Stiff System with Variable Step Sizes.

For the final example, a stiff system was selected to challenge the error estimation procedure and demonstrate its effectiveness for step size control. Briefly stated, a stiff system typically requires exceedingly small time steps to resolve with a stability-limited explicit method, even though the solution may appear smooth. Often, both slow and fast responses (transients) are present. Fixed-point iteration typically struggles to converge [77]. Further explanation may be obtained from Refs. [3] and [49], which both discuss the example selected here. It features a rapid initial transient followed by a slower response that loosely follows $y = cos(t)$. The first-order ODE is

$$y' = -100\,(y - \cos(t))\,, \quad t \in [0, \pi/2] \tag{125}$$

from which the acceleration and jerk constraints can be formulated in terms of v and t.

$$a = -100v - 100\sin(t) \tag{126}$$

$$j = 10000v + 10000\sin(t) - 100\cos(t) \tag{127}$$

The fixed-point iteration algorithm used so far in this paper required the domain to be divided into at least 100 time steps to converge, and the accuracy was still poor. Fortunately, the system of equations is linear with respect to velocity and the acceleration and jerk terms can be eliminated by substitution. The time-dependent, nonlinear forcing terms are lumped into the vector $\mathbf{b}$, resulting in the system

$$\begin{bmatrix} \frac{60}{\Delta t^3} & -\frac{24}{\Delta t^2} - \frac{300}{\Delta t} \\ \frac{60}{\Delta t^3} & -\frac{36}{\Delta t^2} - \frac{900}{\Delta t} - 10000 \end{bmatrix} \begin{Bmatrix} u^1 \\ v^1 \end{Bmatrix} = \tag{128}$$

$$\begin{bmatrix} \frac{60}{\Delta t^3} & \frac{36}{\Delta t^2} - \frac{900}{\Delta t} + 10000 \\ \frac{60}{\Delta t^3} & \frac{24}{\Delta t^2} - \frac{300}{\Delta t} \end{bmatrix} \begin{Bmatrix} u^0 \\ v^0 \end{Bmatrix} + \left\{\mathbf{b}(t^0, t^1)\right\} \tag{129}$$

where the forcing vector is

$$\mathbf{b}(t^0, t^1) = \begin{Bmatrix} \left(-\frac{900}{\Delta t} + 10000\right)\sin(t^0) - 100\cos(t^0) + \frac{300}{\Delta t}\sin(t^1) \\ -\frac{300}{\Delta t}\sin(t^0) + \left(\frac{900}{\Delta t} + 10000\right)\sin(t^1) - 100\cos(t^1) \end{Bmatrix} \tag{130}$$

For each new time step, t^0, t^1, and Δt are known; the coefficient matrices can be calculated once per simulation (or per time step if Δt is varied), and the vector $\mathbf{b}(t^0, t^1)$ can be calculated once per time step. The resulting linear system can be solved directly.

First, the problem was solved using fifty uniform time steps of $\Delta t = \pi/100$. The results are shown in Figure 28(a). The dots are nodal values of the present method. The solid red line is the benchmark, a converged solution produced by the Scientific Python (SciPy) function "`scipy.integrate.odeint`" [65]. The SciPy function uses variable-order Adams and backward difference formula (BDF) routines from the FORTRAN library "`odepack`" and can accommodate stiff systems. The cumulative local error estimates are displayed as bands above and below the present model's solution.

The model clearly overshoots the initial level-off before following the ODE contours parallel to the benchmark solution. By itself the model's solution is misleading – it is qualitatively correct but quantitatively inaccurate. Using the benchmark as the truth, the global error at the last point was 0.135.

The error estimation process successfully detected the struggle of the model in following the rapid transient and widened the error band appropriately. As the solution settled, the local error estimates decreased significantly (in other words, the error band width did not shrink or expand as time progressed). The benchmark solution happens to lie within the cumulative local error bands, though this may not always be the case, and the cumulative local error bands should not be construed as global error bounds. In the absence of the reference solution, the wide error band would suggest to the user that the solution is not trustworthy.

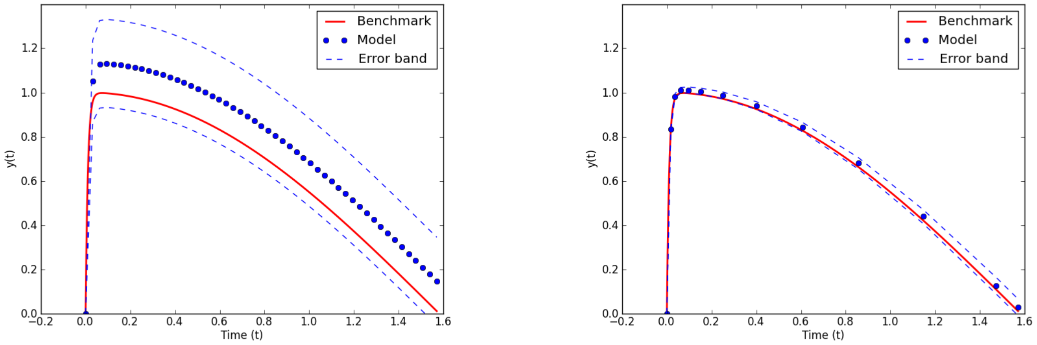
Besides suggesting the solution is inaccurate, the error estimates can also be used to vary the step sizes. To demonstrate this utility, the problem was re-run with step size control performed by a simple rule from Ref. [117]. Let ϵ be the user-defined upper bound for the local truncation error. The local error estimate δu is obtained from Eq. (114) and measured using the maximum norm. After a time step is completed, a new step size is calculated using the formula

$$\Delta t_{new} = \left(\frac{\epsilon}{\|\delta u\|} \right)^{1/5} \Delta t_{previous} \tag{131}$$

If the tolerance for the current step is violated, the new step size is applied to the current step and it is re-calculated. If the tolerance is not violated, the current step solution is accepted and the new step size is applied to the next step.

The solution with variable step sizes is shown in Figure 28(b). An error tolerance of 0.01 was used for accepting a time step's solution; fairly loose, but effective for displaying the new error estimate lines on the plot. The model repeated a step only once (the first time step) and appropriately grouped the points at the initial peak. Only twelve steps were performed and the error was reduced from 0.135 to 0.0175. Again, the benchmark happens to lie within the representative error bands. More significantly, the reported errors are significantly smaller than in the first example, signalling a more accurate solution.

A final case was run with a step size error tolerance of 10^{-8}. The last point's global error was 2.17×10^{-6} with about the same number of steps (52) as the constant step size run (50). On four occasions during the simulation, the algorithm estimated the local error to be out of tolerance, and therefore the time step was repeated with a smaller step size. The local error estimate lines were indistinguishable from the benchmark solution so no plot is shown.



(a) Constant time steps

(b) Variable time steps

Figure 28. Stiff system: Solution and error bands compared to the benchmark.

Summary.

This section has demonstrated the effectiveness of solving initial value problems with quintic Hermite polynomial interpolations defined by end-point jerk constraints. The Newmark methods were shown to be a subset of the current formulation. Complete analysis was performed with the jerk constraints defined according to the governing equation. The method is superior to the second-order Newmark methods in terms of absolute accuracy, rate of convergence, and frequency reproduction. The conditional stability was fully characterized using a linear oscillator, and it was found that the regions of instability existed only above the Nyquist frequency – well above the time step sizes demanded by accuracy requirements. Nonlinear systems are readily accommodated. The problem formulation is systematic and physically intuitive, and can be efficiently executed in a fixed-point iteration algorithm for non-stiff systems.

The accuracy and application of the method has been emphasized, not the computational efficiency. During the course of the study and in initial testing of the nonlinear membrane model, the Hermite interpolation method solved by the fixed-point iteration algorithm has been suitably efficient. Considering the accuracy attained versus the computational expense (both function calls and run time), it is believed that this method can be competitive with other time integration techniques.

Point Collocation Spatial Discretization

In this section, a novel membrane model based on the group finite element formulation and point collocation method will be presented and evaluated for static membranes. After summarizing the governing equations and the material model, polygon interpolation formulas will be derived to calculate gradients in a staggered grid approach. The steps for calculating nodal force imbalances will be described in detail. After the model is explained, verification will demonstrate consistency and an observed rate of convergence of two. Finally, predictions will be validated against experimental results in the literature to show the model to be suitable through its range of intended use (*i.e.*, short of the onset of hyperelastic material response).

Governing Equations.

For this study, a membrane is defined as a thin plate without bending stiffness [63]. The governing partial differential equations (PDEs) for the nonlinear membrane are those of a plate undergoing finite deformations as derived in [111], but with the moment and curvature terms removed. The internal stress components are N_{ij} (second Piola-Kirchoff, per unit length) with the subscripts denoting orientation with respect to the Cartesian axis directions x_1 and x_2. The external pressure vector components are represented by f. The terms are expressed in force per unit area.

$$0 = \frac{\partial N_{11}}{\partial x_1} + \frac{\partial N_{12}}{\partial x_2} + f_1 \tag{132}$$

$$0 = \frac{\partial N_{22}}{\partial x_2} + \frac{\partial N_{12}}{\partial x_1} + f_2 \tag{133}$$

$$0 = \frac{\partial}{\partial x_1}\left(N_{11}\frac{\partial u_3}{\partial x_1} + N_{12}\frac{\partial u_3}{\partial x_2}\right) + \frac{\partial}{\partial x_2}\left(N_{22}\frac{\partial u_3}{\partial x_2} + N_{12}\frac{\partial u_3}{\partial x_1}\right) + f_3 \tag{134}$$

The solution of the membrane surface displacements $\mathbf{u}$ begins by recovery of the surface gradients. The gradients then lead to the components of the Green strain tensor E_{ij} [111, 15].

$$\begin{aligned}
E_{11} &= \frac{\partial u_1}{\partial x_1} + \frac{1}{2}\left[\left(\frac{\partial u_1}{\partial x_1}\right)^2 + \left(\frac{\partial u_2}{\partial x_1}\right)^2 + \left(\frac{\partial u_3}{\partial x_1}\right)^2\right] \\
E_{22} &= \frac{\partial u_2}{\partial x_2} + \frac{1}{2}\left[\left(\frac{\partial u_1}{\partial x_2}\right)^2 + \left(\frac{\partial u_2}{\partial x_2}\right)^2 + \left(\frac{\partial u_3}{\partial x_2}\right)^2\right] \\
E_{12} &= \frac{1}{2}\left[\frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2} + \left(\frac{\partial u_1}{\partial x_1}\frac{\partial u_1}{\partial x_2}\right) + \left(\frac{\partial u_2}{\partial x_1}\frac{\partial u_2}{\partial x_2}\right) + \left(\frac{\partial u_3}{\partial x_1}\frac{\partial u_3}{\partial x_2}\right)\right]
\end{aligned} \tag{135}$$

The internal stresses are calculated using the conventional plane stress constitutive relationship. Prestress is accounted for by the vector $\mathbf{N}_0$. Note that the membrane thickness h, where $h = h(\mathbf{u})$, is a function of the displacement field to take thinning into account as the membrane stretches.

$$\left\{\begin{array}{c} N_{11} \\ N_{22} \\ N_{12} \end{array}\right\} = \frac{Eh\left(\mathbf{u}\right)}{1-\nu^2}\left[\begin{array}{ccc} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \left(1-\nu\right)/2 \end{array}\right]\left\{\begin{array}{c} E_{11} \\ E_{22} \\ 2E_{12} \end{array}\right\} + \left\{\begin{array}{c} N_{11} \\ N_{22} \\ N_{12} \end{array}\right\}_0 \tag{136}$$

Discretization and Polygon Interpolation Formulas.

By using the group formulation, all three governing PDEs (Eq.(132), Eq.(133), and Eq.(134)) were cast into the same first-order PDE form

$$\frac{\partial \mathbf{Q}_1}{\partial x_1} + \frac{\partial \mathbf{Q}_2}{\partial x_2} + \mathbf{f} = 0 \tag{137}$$

where the vectors $\mathbf{Q}$ define the degrees of freedom as

$$\mathbf{Q}_1 = \left\{ \begin{array}{c} N_{11} \\ N_{12} \\ N_{11}\frac{\partial u_3}{\partial x_1} + N_{12}\frac{\partial u_3}{\partial x_2} \end{array} \right\} \qquad \mathbf{Q}_2 = \left\{ \begin{array}{c} N_{12} \\ N_{22} \\ N_{22}\frac{\partial u_3}{\partial x_2} + N_{12}\frac{\partial u_3}{\partial x_1} \end{array} \right\} \tag{138}$$

The domain is discretized by forming a staggered mesh, meaning different variables are evaluated at different points in the domain. For example, the displacements u and the stresses N will not be computed at the same nodes. As commonly found in finite difference discretizations, a staggered mesh enables more compact stencils. In certain CFD applications, high-frequency oscillations are reduced because the pressure and velocity fields are fully coupled [55].

The staggered mesh consists of a primary mesh of three-node linear triangles and a dual mesh of polygons. The nodes of the primary mesh define the model's collocation points and carry the vectors $\mathbf{u}$ (displacement) and $\mathbf{f}$ (external force). The role of the primary mesh is to recover the first partial derivatives of the membrane surface. The calculated partials of each triangle are placed at the centroid, as is common in post-processing gradient recovery procedures [28, 146, 98].

The dual mesh is formed by connecting the centroids of the triangles to form vertex-centered polygons, also called tributary areas [146]. Note that the polygons do not overlap and are therefore not the same as an element patch. The vertices of the polygons carry the $\mathbf{Q}$ vectors. The solution of the discretized governing equations, Eq. (137), requires an approximation of the partial derivatives of $\mathbf{Q}$ at the center node. For this study, the polygon patch interpolation presented in [29] was used. This interpolation is based on the linear interpolation along the edge between adjacent nodes. The shape functions are rational polynomials that interpolate a linear field exactly. For completeness, we first list a few key equations from [29], and then derive the necessary partial derivatives of the shape functions.

Let n be the number of vertices of the polygon and i be an index of those n sides. To use the subscripts to indicate the pertinent node, the coordinates (x_1,x_2) used so far will be renamed (x,y) in this section. Each edge segment l is described by a line with constant coefficients a and b.

$$l_i = 1 - a_i x - b_i y \tag{139}$$

Each polygon will have the center node as its local origin. The coefficients a and b are then determined by the vertex coordinates in the reference configuration.

$$\begin{aligned} a_i &= \frac{y_i - y_{i-1}}{x_{i-1}y_i - x_i y_{i-1}} \\ b_i &= \frac{x_{i-1} - x_i}{x_{i-1}y_i - x_i y_{i-1}} \end{aligned} \tag{140}$$

The line coefficients a and b and the vertex coordinates are used to calculate the relative weight coefficients κ, which are normalized by setting κ_1 to a value of one.

$$\kappa_i = \kappa_{i-1}\left[\frac{a_{i+1}\left(x_{i-1} - x_i\right) + b_{i+1}\left(y_{i-1} - y_i\right)}{a_{i-1}\left(x_i - x_{i-1}\right) + b_{i-1}\left(y_i - y_{i-1}\right)}\right]; \quad \kappa_1 = 1 \tag{141}$$

The line equations and coefficients form the terms of ψ_i, the numerator of the shape function associated with perimeter node i.

$$\psi_i(x, y) = \kappa_i \prod_{\substack{j \neq i \\ j \neq i+1}}^{j=n} l_j(x, y) \tag{142}$$

The denominator polynomial is equal to the sum of all of the numerator polynomials, thus forming the rational polynomial shape function N_i for each node i on the perimeter of the polygon.

$$N_i(x,y) = \frac{\psi_i}{\sum\limits_{j=1}^{n} \psi_j} \tag{143}$$

Having summarized the work of [29], we now obtain the derivatives of the shape functions for use in the present method. Application of the quotient rule leads to the expressions for the shape function derivatives,

$$\frac{\partial N_i}{\partial x}(x,y) = \frac{\left(\sum\limits_{j=1}^{n} \psi_j\right) \frac{\partial \psi_i}{\partial x} - \psi_i \frac{\partial}{\partial x} \left(\sum\limits_{j=1}^{n} \psi_j\right)}{\left(\sum\limits_{j=1}^{n} \psi_j\right)^2} \tag{144}$$

$$\frac{\partial N_i}{\partial y}(x,y) = \frac{\left(\sum\limits_{j=1}^{n} \psi_j\right) \frac{\partial \psi_i}{\partial y} - \psi_i \frac{\partial}{\partial y} \left(\sum\limits_{j=1}^{n} \psi_j\right)}{\left(\sum\limits_{j=1}^{n} \psi_j\right)^2} \tag{145}$$

Each of the terms necessary for this calculation will now be simplified. Multiple applications of the chain rule to Eq. (142) result in the expressions

$$\frac{\partial \psi_i}{\partial x} = \kappa_i \sum_{\substack{k \neq i \\ k \neq i+1}}^{k=n} \left(\frac{\partial l_k}{\partial x} \prod_{\substack{j \neq i \\ j \neq i+1 \\ j \neq k}}^{j=n} l_j \right) \tag{146}$$

$$\frac{\partial \psi_i}{\partial y} = \kappa_i \sum_{\substack{k \neq i \\ k \neq i+1}}^{k=n} \left(\frac{\partial l_k}{\partial y} \prod_{\substack{j \neq i \\ j \neq i+1 \\ j \neq k}}^{j=n} l_j \right) \tag{147}$$

The partial derivatives of the linear edge functions are constants,

$$\begin{aligned} \frac{\partial l_i}{\partial x} &= -a_i \\ \frac{\partial l_i}{\partial y} &= -b_i \end{aligned} \tag{148}$$

which when substituted into Eqs. (146) and (147) reduce them to

$$\begin{aligned} \frac{\partial \psi_i}{\partial x} &= -\kappa_i \sum_{\substack{k \neq i \\ k \neq i+1}}^{k=n} \left(a_k \prod_{\substack{j \neq i \\ j \neq i+1 \\ j \neq k}}^{j=n} l_j \right) \\ \frac{\partial \psi_i}{\partial y} &= -\kappa_i \sum_{\substack{k \neq i \\ k \neq i+1}}^{k=n} \left(b_k \prod_{\substack{j \neq i \\ j \neq i+1 \\ j \neq k}}^{j=n} l_j \right) \end{aligned} \tag{149}$$

Since the local origin was placed at the center node, and the center node is the only point at which this interpolation will be applied, all of the non-constant terms of the derivative shape functions can be disregarded. Then the calculations for the individual terms of Eqs. (144) and (145) reduce to

$$\begin{aligned}
\psi_i &= \kappa_i && (150)\\
\sum_{j=1}^{n}\psi_j &= \sum_{j=1}^{n}\kappa_j && (151)\\
\frac{\partial\psi_i}{\partial x} &= -\kappa_i \sum_{\substack{k\neq i\\ k\neq i+1}}^{k=n} a_k && (152)\\
\frac{\partial\psi_i}{\partial y} &= -\kappa_i \sum_{\substack{k\neq i\\ k\neq i+1}}^{k=n} b_k && (153)\\
\frac{\partial}{\partial x}\left(\sum_{j=1}^{n}\psi_j\right) &= \sum_{j=1}^{n}\frac{\partial\psi_j}{\partial x} && (154)\\
\frac{\partial}{\partial y}\left(\sum_{j=1}^{n}\psi_j\right) &= \sum_{j=1}^{n}\frac{\partial\psi_j}{\partial y} && (155)
\end{aligned}$$

These formulas are substituted into Eqs. (144) and (145) to provide the derivatives of the polygonal shape functions at the center nodes. Thus letting Q_i be scalar values at the polygon vertices $i = 1...n$, the partial derivatives of Q at the polygon's center node are

$$\frac{\partial Q}{\partial x} = \sum_{i=1}^{n}\frac{\partial N_i}{\partial x}Q_i \qquad (156)$$

$$\frac{\partial Q}{\partial y} = \sum_{i=1}^{n}\frac{\partial N_i}{\partial y}Q_i \qquad (157)$$

To improve computational efficiency, much like global stiffness matrices, the individual element interpolation functions are assembled into global derivative matrices $\mathbf{T}_\alpha \equiv \partial/\partial x_\alpha$, where $\alpha = 1, 2$ to indicate the direction of the partial derivative. To distinguish between the sets of matrices, let the superscripts indicate the vector upon which the matrix operates ($\mathbf{u}$ or $\mathbf{Q}$) and the location of the result (c for triangle centroid or n for the interior node of a polygon). The derivative matrices $\mathbf{T}_\alpha^{uc}$ recover the linear triangle partial derivatives from the nodal displacements. The equations for the gradients of linear triangles may be found in most introductory finite element texts. The derivative matrices $\mathbf{T}_\alpha^{un}$ use the polygon interpolations to recover the partial derivatives of $\mathbf{u}$ at the polygon interior nodes from the displacements at the perimeter nodes. Lastly, $\mathbf{T}_\alpha^{Qn}$ use the polygon interpolations to recover the partial derivatives of $\mathbf{Q}$ at the polygon interior nodes from the values $\mathbf{Q}$ at the triangle centroids.

Residual Calculations.

The residual vector $\mathbf{R}$ contains the imbalance between the internal and external forces at the nodes for an approximate solution of the governing equations, Eq. (137). The role of the solver is to reduce the size of the residual vector to an acceptable level with the user supplying the metric and tolerance. Using the entire domain, let n_e be the number of triangular elements, n be the total number of nodes, and n_n be the number of interior nodes. The following quantities are calculated sequentially to return the residual vector $\mathbf{R}$ to the solver.

1. Partial derivatives of $\mathbf{u}$ at the triangle centroids using the primal mesh. Six operations are required (partials of three $\mathbf{u}$ vectors in two directions) using the two pre-calculated discrete derivative operators $\mathbf{T}^{uc}$.

$$\frac{\partial \mathbf{u}_i}{\partial x_\alpha} = \mathbf{T}_\alpha^{uc}\,\mathbf{u}_i \quad \alpha = 1, 2, \quad i = 1, 2, 3 \tag{158}$$

2. Surface unit normal vector $\hat{\mathbf{n}}$ and area ratio J_A at the center nodes. The surface normals at the nodes were obtained via the cross-product of two tangent

vectors [140, 15]. The nodal partial derivatives required by the tangent vectors can be obtained by gradient recovery techniques or by applying the polygon interpolation equations (144) and (145) to the full element patch. The latter technique was applied using the two pre-calculated constant matrices $\mathbf{T}^{un}$ in six operations,

$$\frac{\partial \mathbf{u}_i}{\partial x_\alpha} = \mathbf{T}^{un}_\alpha \, \mathbf{u}_i \quad \alpha = 1, 2, \quad i = 1, 2, 3 \tag{159}$$

Thus at each interior node the tangent vectors were

$$\begin{aligned} \mathbf{g}_1 &= \left\{ 1 + \frac{\partial u_1}{\partial x_1}, \frac{\partial u_2}{\partial x_1}, \frac{\partial u_3}{\partial x_1} \right\}^T \\ \mathbf{g}_2 &= \left\{ \frac{\partial u_1}{\partial x_2}, 1 + \frac{\partial u_2}{\partial x_2}, \frac{\partial u_3}{\partial x_2} \right\}^T \end{aligned} \tag{160}$$

Their cross product's magnitude, J_A, is the ratio of the deformed area to the undeformed area. The direction provides the components of the external force vector. The unit normal vector is

$$\hat{\mathbf{n}} = \frac{\mathbf{g}_1 \times \mathbf{g}_2}{\|\mathbf{g}_1 \times \mathbf{g}_2\|} = \frac{\mathbf{g}_1 \times \mathbf{g}_2}{J_A} \tag{161}$$

3. External force $\mathbf{f}(\mathbf{u})$ at the center nodes. Let p be the spatially-constant magnitude of inflation pressure that acts normal to the membrane in its current configuration. The external force vector is

$$\mathbf{f} = p \, J_A \, \hat{\mathbf{n}} \tag{162}$$

4. Thickness ratio J_h at the center nodes. Incompressibility is assumed for the deformed thickness calculation. Since the volume $V = Ah$ is constant, the

thickness ratio J_h is the reciprocal of the area ratio.

$$J_h = \frac{h}{h_0} = \frac{V/A}{V/A_0} = \frac{A_0}{A} = \frac{1}{J_A} \tag{163}$$

5. $\mathbf{Q}_1$ and $\mathbf{Q}_2$ at the triangle centroids using Eqs. (135), (136), and (138). The deformed thickness $h(\mathbf{u})$ in Eq. (136) will need to be replaced by $J_h h_0$. Since J_h will be calculated at the center node, not at the element centroids where $\mathbf{Q}$ exists, the undeformed membrane thickness h_0 is used for now. The membrane thickness will be corrected in the last step, when the residual vector is calculated.

6. Partial derivatives of $\mathbf{Q}_1$ and $\mathbf{Q}_2$ at the polygons' center nodes by applying the two pre-calculated derivative operators $\mathbf{T}^{Qn}$ to the three $\mathbf{u}$ vectors.

$$\frac{\partial \mathbf{Q}_\beta}{\partial x_\alpha} = \mathbf{T}^{Qn}_\alpha \, \mathbf{Q}_\beta \qquad \alpha, \beta = 1, 2 \tag{164}$$

7. Global residual vector $\mathbf{R}$. The nodal thickness ratio J_h from Eq. (163) is now applied to the partial derivatives of $\mathbf{Q}$ to calculate $\mathbf{R}_j \in \Re^3$, residual vector of node j. The nodal residuals are assembled into the global residual vector $\mathbf{R} \in \Re^{3n_n}$, which is returned to the nonlinear solver for further minimization.

$$\mathbf{R}_j = \left(J_h \frac{\partial \mathbf{Q}_1}{\partial x_1} + J_h \frac{\partial \mathbf{Q}_2}{\partial x_2} + \mathbf{f} \right)_j \qquad j = 1 \ldots n_n \tag{165}$$

In the point collocation method, the residual equations for points on the boundary are determined by the boundary conditions and will be different from the domain interior's governing equation [1]. Here, for the homogeneous boundary conditions, the displacements are simply set to zero, and the boundary nodes are removed from the solution vector.

Nonlinear Solver.

The dynamic simulation under development which will use this membrane model primarily employs an accelerated fixed-point iteration algorithm. This method does not require the construction of a Jacobian matrix (or even an approximation of one), so significant savings in computational effort are expected. Since the Newton-Raphson method will not be used, the tangent stiffness matrix is not needed and will not be formulated analytically. Thus for the static cases presented here it was necessary to find an efficient alternative method that estimates the Jacobian or avoids using it altogether.

For this study we utilized “`scipy.optimize.newton_krylov`”, the Newton-Krylov nonlinear equation solver from the Python library SciPy [65]. The solver was supplied an initial guess for $\mathbf{u}$ and a callable function that returned the residual vector $\mathbf{R}(\mathbf{u})$, Eq. (165). The loose generalized minimum residual (LGMRES) method was selected as the inner solver [6].

Broadly speaking, Newton-Krylov methods employ nested iterative solvers. The outer solver performs corrections like the classical Newton method. The inner solver is one of many linear Krylov subspace methods [131]. Jacobian-free Newton-Krylov (JFNK) methods like the one used in this SciPy routine are efficient for large systems because they use a perturbation of the entire solution vector to approximate Jacobian-vector products; this approach is more efficient than the finite difference Jacobian construction, which requires a perturbation of each element of the solution vector [75, 69]. The actual Jacobian is not needed, yet convergence can approach that of the Newton method. Preconditioning by providing an approximation to the Jacobian is highly recommended and often necessary for adequate performance [12, 131]. However, for the simulations in this study, performance was more than adequate without supplying a preconditioner to the solver.

Because stress stiffening is the source of transverse resistance, the solver will fail when starting from a flat, slack membrane. There are several remedies for this problem. The approach used in this study was to prescribe an initially non-flat shape ([141] used this approach for a box-shaped membrane and mentioned its necessity). Simple parabolic profiles were sufficient. A second option, dynamic relaxation, uses a dynamic model with damping to settle to the static solution [64, 28, 140]. This option is convenient since it is not necessary to code a separate solver for the static solution; and it also has the benefit of verifying some aspects of the dynamics code. However, convergence can be extremely slow if the damping mechanism is not carefully designed. Some of these challenges can be avoided by using pseudo-transient continuation [40, 70]. The non-physical time step sizes can be controlled using the successive evolution-reaction (SER) technique [75]. A third approach, also not employed herein, entails initially applying a pretension to aid the solver and later removing it for the final solution.

It is worth mentioning a few other numerical alternatives that may be used in the absence of an analytical Jacobian. A more comprehensive review of these alternatives may be found in [140]. Finite difference Jacobian approximations are easy to perform but are not robust, and they become very expensive as the size of the problem increases. Automatic differentiation extracts the derivatives directly from the code [47]. Multigrid methods [21, 52] (in particular the Full Approximation Scheme [20]), and combinations of Newton-Krylov and multigrid methods [66, 41] require some careful coding, but the ultimate computational efficiency gains can be impressive. Lastly, depending upon how the method is formulated, dynamic relaxation may also be a viable alternative.

Verification.

Patch tests were performed for the linear plane stress scenario [58]. All of the tests featured a unit vertical rigid body translation. The square patch measured two units per side and was centered at $(1, 1)$ as shown in Figure 29. The patch contained a single interior node at $(1.6, 1.4)$, approximately on the perimeter of its dual-mesh polygon. Given a linear displacement field, boundary nodes were displaced accordingly and the displacement of the center node was checked against the exact field. Introducing Young's modulus E, the physical constants were set to $E = 1000$, $h = 1$, and $\nu = 0.5$. The results in Table 7 show that the model exactly reproduced the constant strains and stresses, even with irregular elements.

Further verification and convergence determination was performed using the Method of Manufactured Solutions [112, 23]. In this method, a solution is fabricated (it does not have to be physically plausible), and the forcing function is calculated from the governing PDE. Then, the forcing function is used in the numerical model to obtain an approximate solution. The error is then the difference between the model's solution

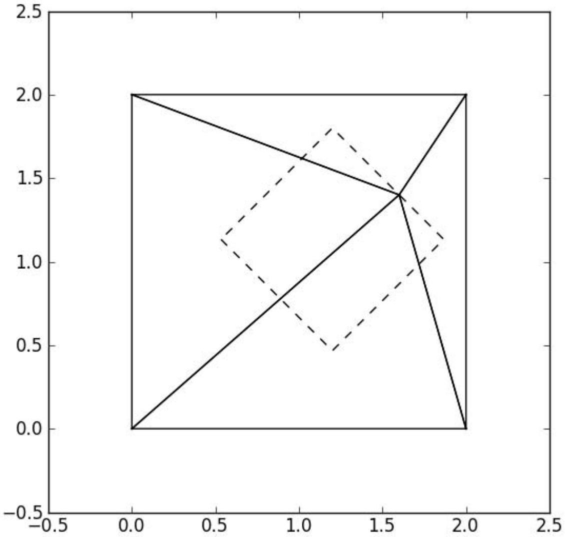


Figure 29. Square 2×2 5-node patch for the patch test. Solid lines are the triangles of the primary mesh; dashed lines illustrate the node-centered polygon of the dual mesh.

Table 7. Patch test results. A star (*) indicates a non-zero magnitude of less than 10^{-11}; strains and stresses for all four elements were reported as exact when integer values, and exact to at least eight decimal places when fractional.

$\mathbf{u}$ field	$\mathbf{u}$(1.6,1.4)	$\mathbf{E}$	$\mathbf{N}/h$
(1,0,1)	(1,0*,1)	(0,0*,0*)	(0*,0*,0*)
(0,1,1)	(0*,1,1)	(0,0*,0*)	(0*,0*,0*)
(x,0,1)	(1.6,0*,1)	(1.5,0*,0)	(2000, 1000, 0)
(0,x,1)	(0*,1.6,1)	(0.5,0*,0.5)	(666.7,333.3,333.3)
(y,0,1)	(1.4,0*,1)	(0,0.5,0.5)	(333.3,666.7,333.3)
(0,y,1)	(0*,1.4,1)	(0,1.5,0*)	(1000, 2000, 0*)

and the manufactured solution. The manufactured solution, Eq. (166), was devised in accordance with the recommended guidelines found in [115]. Non-unity constants were chosen such that the solution magnitudes and derivatives were of approximately the same order of magnitude in all three axes such that potential formula errors might be revealed.

$$\mathbf{u}\left(x_1, x_2\right) = \left\{ \begin{array}{c} 0.17\mathrm{e}^{(x_1/2-x_2/2)} \\ -0.37\mathrm{e}^{(-x_1/4-x_2/2)} \\ 0.71\mathrm{e}^{(x_1+x_2)} \end{array} \right\} \tag{166}$$

The lengthy formulas for the corresponding anisotropic pressure vector were obtained by substituting the manufactured solution $\mathbf{u}$ of Eq. (166) into Eqs. (132), (133), (134), (135), and (136). This process was performed independently of the point collocation code. The pressure vector formulas were generated symbolically with the Python library SymPy [129] and inserted into the point collocation model. The Dirichlet boundary conditions were satisfied by constraining perimeter node displacements in accordance with the manufactured solution, Eq. (166).

To investigate the effects of discretization error, verification was performed on two domains: a hexagon and a circle. The hexagonal domain was ideally discretized with a structured, symmetrical primary mesh of equilateral triangles to minimize discretization error. The six points of the domain laid on the unit circle. The unit-radius circular domain was discretized with asymmetrical, unstructured grids—the same grids which will be used later in the validation phase. Three grids from each domain are shown in Figure 30. All grids were created using the open source software Gmsh [43], which contains a "refine by splitting" feature (elsewhere called "refine by quartering" [74]) to easily perform structured refinement with a grid ratio of two. For this study the representative element size of a mesh, h^e, was calculated as the length of the side of an equilateral triangle, where the area of the triangle was equal to the mean element area. The error measures ϵ_u and ϵ_E were the Euclidean norms of the displacement error vector and the strain error vector at the origin, respectively. The strain was recovered as the mean of the strains of the surrounding triangular elements. Both measures were normalized by the value of the exact solution.

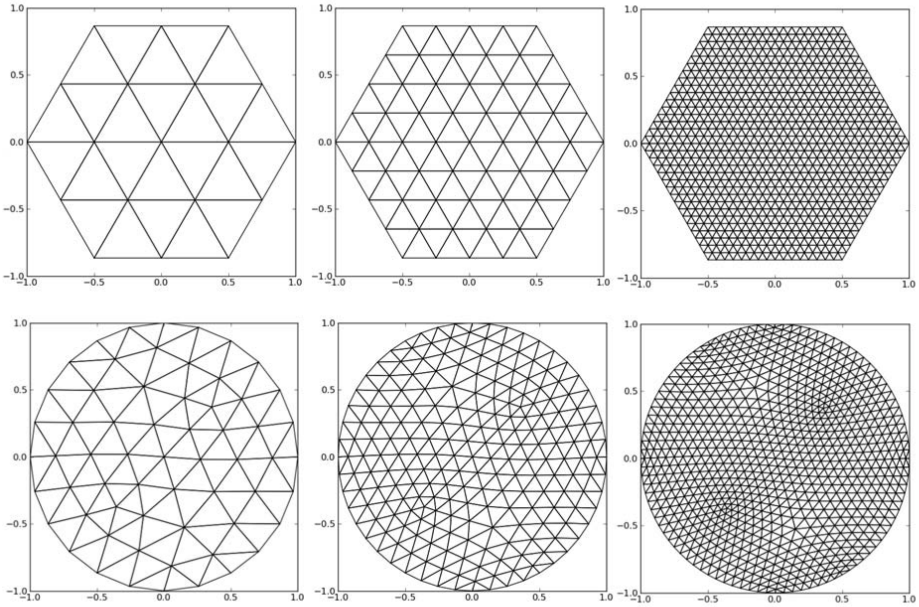


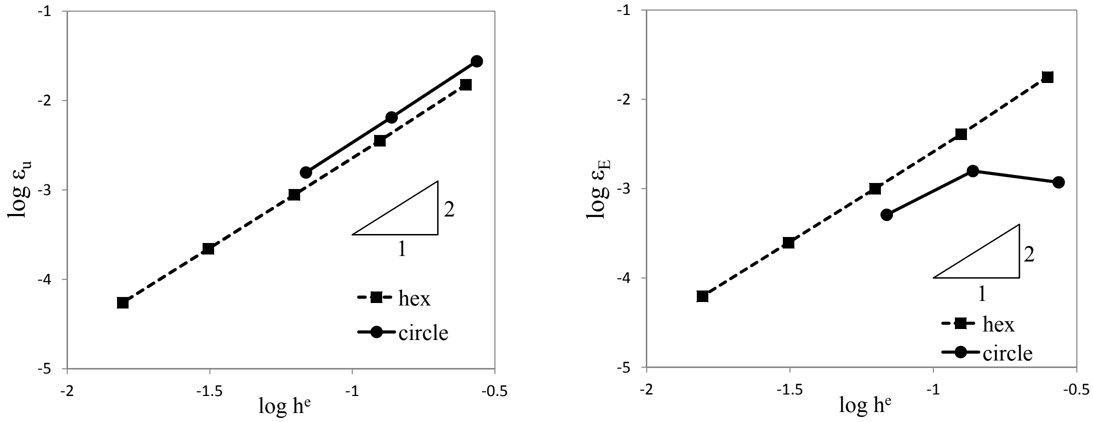
Figure 30. Point collocation static verification and validation meshes. Top row: Hexagonal grids with $h^e = 0.5$, 0.25, and 0.0625 (hex2, hex4, and hex16). Bottom row: The three finest circular grids, used for both verification and validation (circle3, circle2, and circle1, left-to-right).

The convergence study results are shown in Table 8 and Figure 31. The observed order of accuracy p converged to two for both displacement and strain in the hexagonal domain. The displacement errors were slightly higher for the circular grids due to the discretization error of the unstructured grid; however, the same order of convergence was observed. The strain convergence in the circular grid did not behave as neatly because the individual strain vector components converged differently: E_{11} from above, E_{12} from below, and E_{22} non-monotonically. Thus the vector norm converged non-monotonically. The slope of a least-squares linear fit of the four circular grids' strain errors provided $p = 2.24$, more in line with the other convergence rates. Interestingly, the circular grids produced more accurate strain predictions than the hexagonal grids despite the non-monotonic strain convergence behavior and lower displacement accuracy.

Table 8. Convergence study results. The hexagonal grids are labeled with the number of elements from the origin to the vertex at $(1, 0)$. The circular grids are numbered sequentially from fine to coarse.

Grid	h^e	$\log(h^e)$	$\log(\epsilon_u)$	p_u	$\log(\epsilon_E)$	p_E
hex2	0.5	-0.30	-1.04	–	-0.98	–
hex4	0.25	-0.60	-1.74	2.32	-1.75	2.56
hex8	0.125	-0.90	-2.36	2.07	-2.39	2.13
hex16	0.0625	-1.20	-2.97	2.02	-3.00	2.03
hex32	0.03125	-1.51	-3.57	2.00	-3.60	2.01
hex64	0.015625	-1.81	-4.17	2.00	-4.21	2.00
circle4	0.537	-0.27	-0.77	–	-1.02	–
circle3	0.273	-0.56	-1.47	2.39	-2.93	6.50
circle2	0.137	-0.86	-2.10	2.10	-2.80	-0.42
circle1	0.069	-1.16	-2.72	2.05	-3.29	1.63

The verification process has demonstrated that the model correctly and consistently solved the coded governing equations. The method's observed order of convergence of both displacement and strain was two. Also, the circular grids introduced an acceptably small amount of discretization error and are therefore suitable for use in the validation phase.



(a) Displacement (b) Strain

Figure 31. Convergence study. The triangle displays a reference slope of two. Dashed lines are hexagonal grid results; solid lines are circular mesh results. Grids hex2 and circle4 from Table 8 are omitted.

Validation.

The point collocation model was validated against the experimental and finite element model results in [125]. In that study, a latex rubber sheet was inflated from below, and the displacement field was extracted by optically tracking a random speckling pattern on the membrane surface. Strains were calculated from the displacement field during post-processing. A geometrically nonlinear finite element model was compared to another finite element approximation by [122], then validated against the experimental data.

Two cases were selected for validation in this paper: one with and the other without uniform prestress. In the case without prestress, model predictions were compared directly to the experimental data. The physical constants were outer radius $R = 57.15$ mm, thickness $h = 0.12$ mm, modulus of elasticity $E = 2$ MPa, and Poisson ratio $\nu = 0.5$.

In the prestressed case, a non-isotropic prestrain field precluded use of the experimental results. The geometrically nonlinear finite element model was therefore used as a benchmark for this study. The FEM model was suitable as a benchmark because it was validated in [125] against the experimental data for the same problem configuration (including geometry, boundary conditions, material, and load type) and to much greater loads and deformations. The physical constants for this case were $R = 3.5$ mm, $h = 1.0\mu$m, $E = 71.0$ GPa, and $\nu = 0.345$. Following convention for the Hencky problem, the lateral deflection is normalized as w/R, and the nondimensional pressure q normalizes the inflation pressure as $q = pR/Eh$.

Representative grid convergence results are shown in Table 9. The format of the presentation comes from [24], which put forth useful guidelines to standardize the reporting of CFD numerical study results. As explained earlier, all grid ratios are equal to two. The symbol ϕ represents the magnitude of the field variable (displacement or strain). A subscript 1, 2, or 3 indicates the results were obtained from the grid circle1, circle2, or circle3 respectively (see Table 8 and Figure 31). The calculated order of accuracy is p. The symbol ϕ_{ext}^{32} indicates the value was obtained from Richardson extrapolation using grids 2 and 3. The approximate and extrapolated relative errors (magnitude percentage change in ϕ from one grid to the next finer) are shown as ϵ_a and ϵ_{ext}.

The convergence study confirms that the grids were sufficiently refined, as indicated by the convergence of the extrapolated values ϕ_{ext} and the small relative errors ϵ. The extrapolated values ϕ_{ext}^{21} were taken as the model solution for the remaining discussions. The second-order observed accuracy from the verification phase was maintained for both displacement and strain within the range of the model's expected applicability. The order of convergence of w/R deteriorated to below one for large deformations (*i.e.*, for $w/R \gtrsim 0.4$).

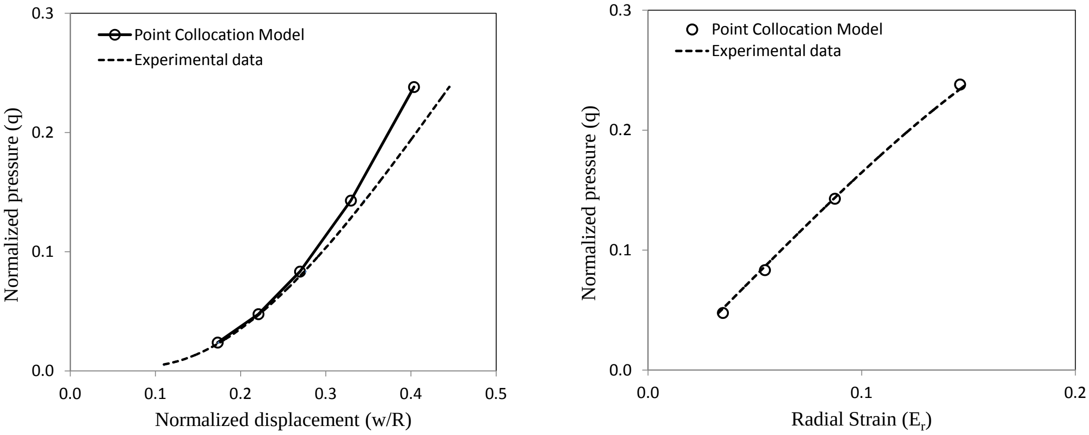
The point collocation model predictions are compared to the experimental results in Figure 32 and to the FEM model benchmark in Figure 33. Agreement is excellent

Table 9. Validation convergence study for displacement (w/R) and radial strain (E_r) at the origin.

Dep. variable ϕ Prestress q	w/R None 0.024	w/R None 0.048	E_r None 0.048	E_r None 0.083	w/R 250MPa 0.0030
ϕ_3(coarse mesh)	0.1748	0.2223	0.0339	0.0525	0.08005
ϕ_2(medium)	0.1739	0.2212	0.0348	0.0541	0.07983
ϕ_1(fine)	0.1737	0.2210	0.0350	0.0546	0.07978
p	2.00	1.96	2.04	1.98	2.05
ϕ_{ext}^{32}	0.1736	0.2209	0.0350	0.0547	0.07976
ϕ_{ext}^{21}	0.1736	0.2209	0.0350	0.0547	0.07976
ϵ_a^{21}	0.13%	0.12%	0.59%	0.76%	0.07%
ϵ_{ext}^{21}	0.04%	0.04%	0.19%	0.26%	0.02%

with and without prestress up to $w/R \approx 0.25$, beyond which the point collocation model begins to underestimate the displacement. This behavior is in agreement with the comparison of Mooney-Rivlin and Hookean material models in [106], so the gradual loss of accuracy beyond this point may be attributed to the onset of hyperelastic effects. The nodal strain predictions continue to match the experimental data beyond $w/R \approx 0.45$. Just as in the verification phase, the strains are actually more accurate than the displacements. Typically the directly-calculated derived variables (strains and stresses) converge more slowly than the displacements, though post-processing recovery procedures may improve the accuracy [146, 98]. The staggered mesh of this model uses constant-strain triangles, so nodal strain values must be recovered by one of the gradient recovery procedures. Simple averaging from neighboring elements was sufficient to produce the excellent relative accuracy of strain at the center node.

Validation against the experimental data has shown that the model is accurate for displacements up to $w/R \approx 0.25$, at which point a hyperelastic model would be more appropriate as discussed in [125].



(a) Displacement

(b) Radial Strain

Figure 32. Validation with no prestress by comparison to experimental data.

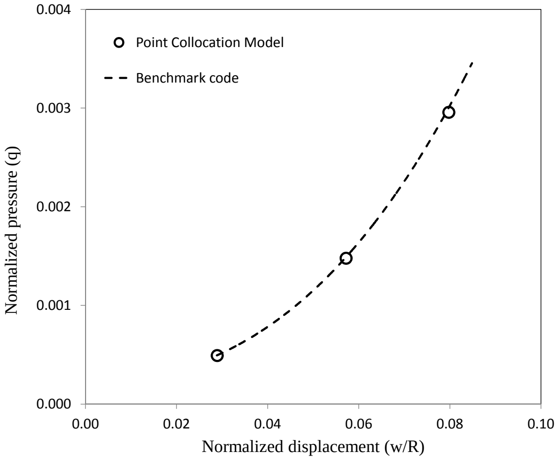


Figure 33. Validation of the prestressed case by comparison to the FEM benchmark.

Summary.

A membrane model intended for eventual use in dynamic aeroelastic simulations was presented in this section, and its performance for membranes at static equilibrium was investigated. The model effectively combines several unconventional formulations in structural engineering, including a staggered grid with robust low-order interpolation schemes, grouped nonlinear products as degrees of freedom, and the point collocation method. Method capabilities include variable thickness, follower forces, and arbitrary prestress. Rigorous verification demonstrated consistency, and the observed order of convergence was two for both displacement and strain. During validation with respect to a static circular membrane (the Hencky problem), the point collocation model predictions agreed with experimental data and benchmark FEM code until the region where hyperelastic response began to dominate.

The primary feature that distinguishes this approach is its simplicity. Element integration is avoided entirely. The group formulation permits the same treatment in all three axes, and the resulting code is explicit and self-documenting. Overall, the framework of the approach is highly modular and flexible. Any given step can be performed by interchangeable subroutines. For example, the polygon interpolation technique could be replaced by least squares or radial basis function routines without replacing the remaining code. The residual subroutine readily accepts different strain-displacement or material models (including nonlinear models).

Membrane Dynamics

This section presents the model that fulfills the objectives of this research by accurately simulating the transient response of a geometrically nonlinear membrane. The model incorporates the new temporal and spatial discretization schemes, namely the Hermite time interpolation method of Section IV and the point collocation method of Section IV.

The complete dynamic membrane model will be verified using the Method of Manufactured Solutions. Rates of convergence will be calculated from the results. A free-response case study will be performed using the converged results of the two time-integration methods as the baseline solution. Throughout this section, results of the model using the Hermite integration method will be compared to results using the second-order Newmark trapezoid method (shortened to “the Newmark method” from now on). Despite the disparity in the orders of the methods, the Newmark method was chosen due to its prevalence and familiarity in the structural engineering community. Additionally, since two different dynamic time scales were observed, a hybrid method will be investigated that uses the Newmark trapezoid method for in-plane dynamics and the Hermite interpolation method for out-of-plane dynamics.

The governing equations for the dynamic nonlinear membrane were derived in [111]. They are identical to those of Section IV (Eq.(132), Eq.(133), and Eq.(134)) but with the inertial forces retained. The density ρ is in terms of mass per unit volume.

$$\rho h \ddot{u}_1 = \frac{\partial N_{11}}{\partial x_1} + \frac{\partial N_{12}}{\partial x_2} + f_1 \tag{167}$$

$$\rho h \ddot{u}_2 = \frac{\partial N_{22}}{\partial x_2} + \frac{\partial N_{12}}{\partial x_1} + f_2 \tag{168}$$

$$\rho h \ddot{u}_3 = \frac{\partial}{\partial x_1}\left(N_{11}\frac{\partial u_3}{\partial x_1} + N_{12}\frac{\partial u_3}{\partial x_2}\right) + \frac{\partial}{\partial x_2}\left(N_{22}\frac{\partial u_3}{\partial x_2} + N_{12}\frac{\partial u_3}{\partial x_1}\right) + f_3 \tag{169}$$

Retaining from Section IV the expressions for strains and stresses, as well as the definitions for the vectors $\mathbf{Q}$, the vectorized version of the dynamic governing equations is therefore

$$\rho h \ddot{\mathbf{u}} = \frac{\partial \mathbf{Q}_1}{\partial x_1} + \frac{\partial \mathbf{Q}_2}{\partial x_2} + \mathbf{f} \tag{170}$$

Jerk Constraints.

The jerk constraints were derived by taking the time derivative of the equilibrium equations, Eq. 170, to obtain

$$\mathbf{j} = \frac{1}{\rho h}\left(\frac{\partial \dot{\mathbf{Q}}_1}{\partial x_1} + \frac{\partial \dot{\mathbf{Q}}_2}{\partial x_2} + \dot{\mathbf{f}}\right) \tag{171}$$

The jerk constraints require formulas for the strain rates of change,

$$\begin{aligned}
\dot{E}_{11} &= \frac{\partial v_1}{\partial x_1} + \left(\frac{\partial u_1}{\partial x_1}\frac{\partial v_1}{\partial x_1}\right) + \left(\frac{\partial u_2}{\partial x_1}\frac{\partial v_2}{\partial x_1}\right) + \left(\frac{\partial u_3}{\partial x_1}\frac{\partial v_3}{\partial x_1}\right) && (172)\\
\dot{E}_{22} &= \frac{\partial v_2}{\partial x_2} + \left(\frac{\partial u_1}{\partial x_2}\frac{\partial v_1}{\partial x_2}\right) + \left(\frac{\partial u_2}{\partial x_2}\frac{\partial v_2}{\partial x_2}\right) + \left(\frac{\partial u_3}{\partial x_2}\frac{\partial v_3}{\partial x_2}\right) && (173)\\
2\dot{E}_{12} &= \frac{\partial v_2}{\partial x_1} + \frac{\partial v_1}{\partial x_2} + \left(\frac{\partial v_1}{\partial x_1}\frac{\partial u_1}{\partial x_2}\right) + \left(\frac{\partial u_1}{\partial x_1}\frac{\partial v_1}{\partial x_2}\right) && (174)\\
&\quad + \left(\frac{\partial v_2}{\partial x_1}\frac{\partial u_2}{\partial x_2}\right) + \left(\frac{\partial u_2}{\partial x_1}\frac{\partial v_2}{\partial x_2}\right) + \left(\frac{\partial v_3}{\partial x_1}\frac{\partial u_3}{\partial x_2}\right) + \left(\frac{\partial u_3}{\partial x_1}\frac{\partial v_3}{\partial x_2}\right) && (175)
\end{aligned}$$

which after following the same process as described above provide the rates of change of $\mathbf{Q}$,

$$\dot{\mathbf{Q}}_1 = \left\{\begin{array}{c} \dot{N}_{11} \\ \dot{N}_{12} \\ \dot{N}_{11}\frac{\partial u_3}{\partial x_1} + N_{11}\frac{\partial v_3}{\partial x_1} + \dot{N}_{12}\frac{\partial u_3}{\partial x_2} + N_{12}\frac{\partial v_3}{\partial x_2} \end{array}\right\} \tag{176}$$

$$\dot{\mathbf{Q}}_2 = \left\{\begin{array}{c} \dot{N}_{12} \\ \dot{N}_{22} \\ \dot{N}_{22}\frac{\partial u_3}{\partial x_2} + N_{22}\frac{\partial v_3}{\partial x_2} + \dot{N}_{12}\frac{\partial u_3}{\partial x_1} + N_{12}\frac{\partial v_3}{\partial x_1} \end{array}\right\} \tag{177}$$

The calculation of the jerk, j, uses much of the same code as that of the acceleration, a. The spatial discretization algorithms and data structures are utilized for both, as well as surface gradients. Thus, in terms of computational expense, where the calculation of a is one function call, the work required to calculate both a and j is less than two function calls. As detailed in Section IV, from classical analysis of a linear oscillator, the resulting scheme is conditionally stable, frequency-preserving, non-dissipative, and more accurate than the Newmark trapezoidal and central difference methods by several orders of magnitude.

Verification.

Performance analysis was accomplished using the Method of Manufactured Solutions [112, 23]. In this procedure, a solution is fabricated (it does not have to be physically plausible) and the forcing function is analytically derived from the governing PDE and the constitutive equations. The forcing functions, having been derived through symbolic differentiation, are discretization-independent. When the simulations are run for a discretized domain, the Dirichlet boundary conditions are set by the manufactured solution and the derived forcing function is applied across the domain. The error is then the difference between the model's solution and the manufactured solution.

The domain was defined to be a hexagon circumscribed by a circle of radius $R = 57.15$ mm (the radius of the circular membrane in Ref. [125]). See Fig. 34 for a depiction. The origin was placed at the center of the domain. This domain contained the structured primary mesh, which consisted of equilateral triangles with side length $h^e = R/8$.

The manufactured solution crafted for this study, Eq. (178), dictated a smooth,

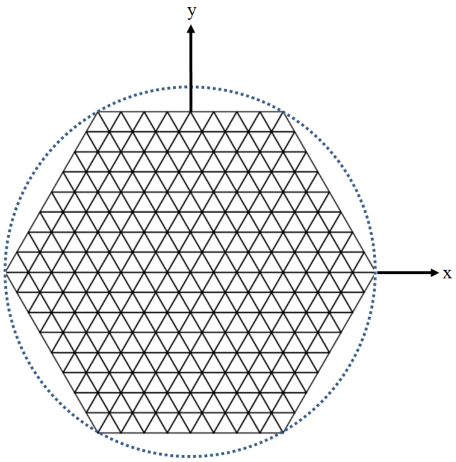


Figure 34. Hexagonal grid for the dynamic membrane verification, showing the circumscribed circle of radius R.

curved surface by the displacement fields,

$$\mathbf{u}\left(x,y,t\right)=A\left(t\right)\begin{Bmatrix} e^{\mathbf{x}} \\ e^{\mathbf{y/2}} \\ e^{\mathbf{x+y}} \end{Bmatrix} \tag{178}$$

where the time-varying amplitudes were

$$A\left(t\right)=\frac{R}{10}\left(\cos 500t+2\right) \tag{179}$$

At the origin, the exact solutions for displacement in all three axes were precisely $A(t)$, so it was known *a priori* that any frequency content where $\omega \neq 500$ rad/sec was induced by model errors. The physical constants were chosen to be those of the experimental membrane in Ref. [125]: membrane thickness $h = 0.12$ mm, modulus of elasticity $E = 2$ MPa, density $\rho = 990$ kg/m^3, and Poisson ratio $\nu = 0.5$. The lengthy formulas for the pressure vectors and their rates of change that produce the solution of Eq. 166 were generated symbolically with the Python library SymPy [129].

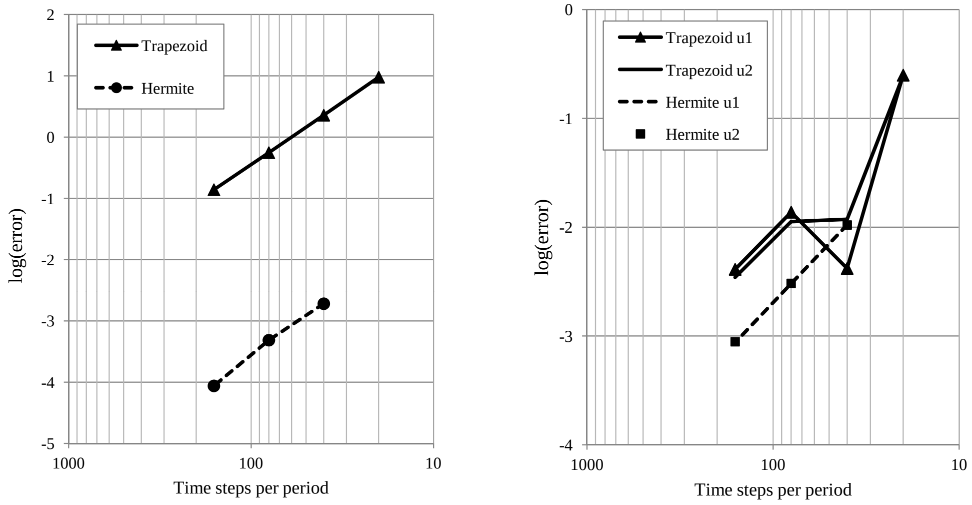
A convergence study was performed to compare the Hermite interpolation method with physical jerk constraints from Eq. 171 to the Newmark trapezoid method, where $\gamma = 0.5$ and $\beta = 0.25$. The simulation was run for one period of the exact solution ($T = 2\pi/500$ s) using 20, 40, 80, and 160 time steps. First, the results with respect to the final displacement errors at the origin will be discussed, followed by results with respect to mean errors across all of the spatial nodes throughout the simulation.

The convergence data using the final displacements u_1, u_2, and u_3 at the origin are listed in Table 10 and plotted in Fig. 35, with the out-of-plane errors shown separately from the in-plane errors. Error was normalized as a percentage of the exact solution's amplitude. The values shown are $\log_{10}|error|$. An asterisk indicates invalid data due to non-monotonic convergence, and "n/a" indicates lack of data due to failure of the algorithm to converge. The rates of convergence, p, are shown for each increment of mesh refinement. The uneven trend lines of the in-plane errors for the trapezoid method were produced by their non-monotonic convergence. The Hermite method improved upon the Newmark's accuracy by three orders of magnitude for out-of-plane errors, but the techniques were comparable for in-plane displacements. The observed rate of convergence was approximately two for both methods.

Table 10. Dynamic nonlinear membrane verification convergence data for the final displacements at the origin.

	Trapezoid			Hermite		
Steps	u_1	u_2	u_3	u_1	u_2	u_3
20	-0.60	-0.61	0.98	n/a	n/a	n/a
40	-2.38	-1.93	0.36	-1.98	-1.98	-2.72
80	-1.86	-1.95	-0.25	-2.52	-2.52	-3.31
160	-2.39	-2.46	-0.86	-3.06	-3.05	-4.06
p 20 → 40	5.90*	4.38*	2.06	n/a	n/a	n/a
p 40 → 80	-1.71*	0.07*	2.03	1.79	1.78	1.98
p 80 → 160	1.74	1.70	2.01	1.80	1.78	2.48

The convergence study results using the mean error magnitude for all spatial



(a) Out of plane

(b) In-plane

Figure 35. Dynamic nonlinear membrane convergence plots for the final displacements at the origin.

nodes over all time steps are shown in Table 11 and plotted in Fig. 36. The error is defined as $error = \sum |u_i - \bar{u}_i| / N^n(N+1)$, where $\bar{u}$ are the exact solutions, N^n is the number of nodes in the mesh, and N is the number of time steps in the simulation. The values shown are $\log_{10} |error|$. The entry "n/a" indicates lack of data due to failure of the algorithm to converge. The rates of convergence, p, are shown for each increment of mesh refinement. The observed rate of convergence for both methods was two, the same as that of the final displacement errors at the origin in Fig. 35(a). With this metric, the accuracy of the Hermite method was approximately 1.5 orders of magnitude better than the trapezoid method.

The reason why the Hermite method's accuracy was significantly greater for out-of-plane displacements than for in-plane displacements is depicted clearly in Fig. 37, a representative illustration of the errors over time from a separate simulation using the trapezoid method. The representative plot of normalized model errors was generated using the Newmark trapezoid method and external forcing for a manufactured solution. It is evident that the oscillation frequency of in-plane errors significantly ex-

Table 11. Dynamic nonlinear membrane convergence data for the mean error magnitudes throughout the simulation.

Steps	Trapezoid	Hermite
20	0.25	n/a
40	-0.35	-1.98
80	-0.96	-2.57
160	-1.56	-3.17
p 20 → 40	2.00	n/a
p 40 → 80	2.00	1.98
p 80 → 160	2.00	1.98

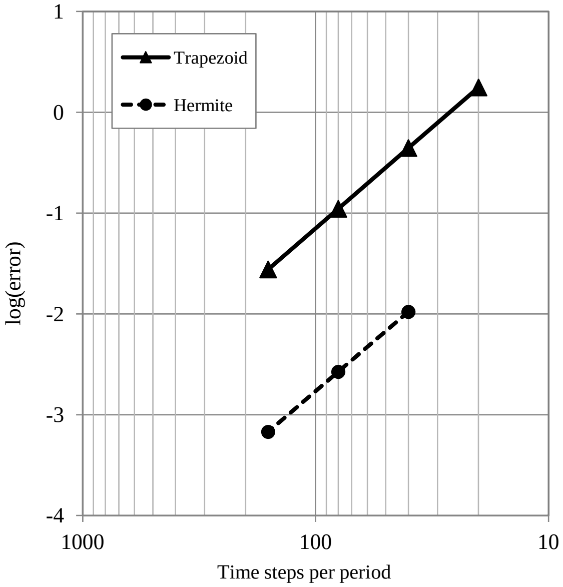


Figure 36. Dynamic nonlinear membrane convergence plot of the mean error magnitudes for all three axes, all spatial nodes, all time steps.

ceeded that of the out-of-plane error. A membrane is much stiffer in-plane, where the tensile internal forces respond linearly with displacement. In contrast, the resistance to out-of-plane motion is zero for a flat membrane, and stress-stiffening occurs with ever greater transverse deformation. Thus the present system has two time scales for an integrator to deal with. The Hermite method's accuracy can be fully realized for the slow signal, where it significantly outperforms the trapezoid method. However, the fast in-plane response can produce oscillatory results or cause the conditionally stable Hermite method to diverge prior to run completion as the time step increases, and in fact the solution did not converge for the time step size $\Delta t/T = 0.05$. (Note that the period T in this calculation is the period of the manufactured solution, not of the numerical solution's oscillations as depicted in Fig. 37.)

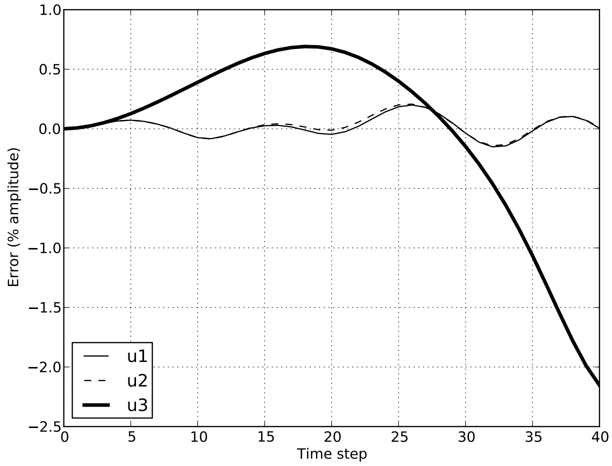


Figure 37. Illustration of the nonlinear membrane's two time scales: rapid in-plane responses (u_1 and u_2), and slower out-of-plane dynamics (u_3).

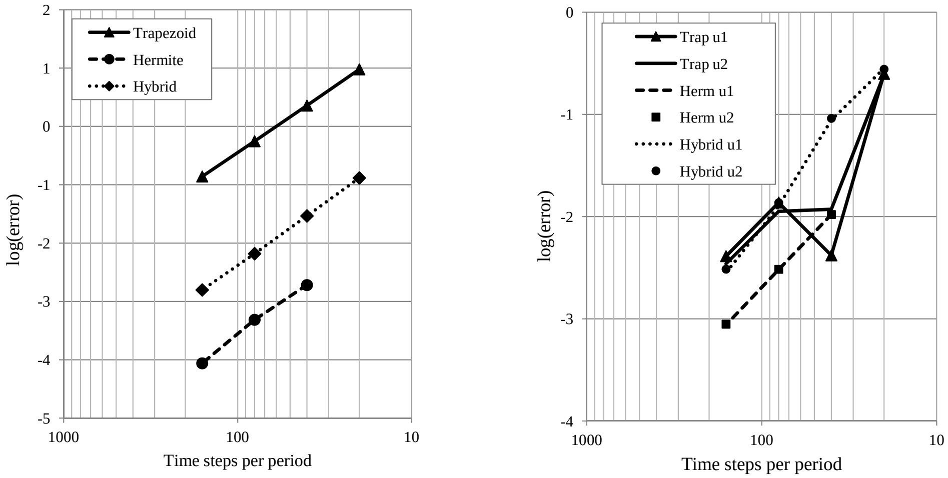
Following these observations, a hybrid technique was explored to capitalize on the strengths of both the Newmark and the Hermite methods: the unconditional stability of the trapezoid method for the rapidly changing in-plane displacements, and the significantly improved accuracy of the Hermite method for the slower out-of-plane dynamics. It was proved in Section IV that the Newmark could be obtained by setting the jerk constraints appropriately. Therefore, coding was straight-forward to utilize the different jerk constraints for in-plane versus out-of-plane equations of motion.

The results of the previous convergence studies are now presented again, but with the hybrid method included for comparison. The convergence data using the final displacements at the origin are listed in Table 12 and plotted in Fig. 38 (with data carried forward from Table 10 and Fig. 35). The convergence study results using the mean error magnitude for all spatial nodes over all time steps are shown in Table 13 and plotted in Fig. 39 (with data carried forward from Table 11 and Fig. 36).

As expected, the convergence data show that the hybrid method effectively combined the strengths of the Hermite and Newmark methods. In-plane errors were comparable to the pure trapezoid method, but the out-of-plane accuracy was better than the trapezoid method by two orders of magnitude as shown in Fig. 38(a). Using the mean error norm, the hybrid method errors were an order of magnitude smaller than those of the trapezoid method. In addition, as far as the improving stability, the hybrid method successfully completed the simulation with time step size $\Delta t/T = 0.05$, where the purely Hermite method failed.

Table 12. Dynamic nonlinear membrane verification convergence data for the final displacements at the origin, including the hybrid method.

	Trapezoid			Hermite			Hybrid		
Time Steps	u_1	u_2	u_3	u_1	u_2	u_3	u_1	u_2	u_3
20	-0.60	-0.61	0.98	n/a	n/a	n/a	-0.55	-0.56	-0.88
40	-2.38	-1.93	0.36	-1.98	-1.98	-2.72	-1.06	-1.04	-1.53
80	-1.86	-1.95	-0.25	-2.52	-2.52	-3.31	-1.90	-1.86	-2.18
160	-2.39	-2.46	-0.86	-3.06	-3.05	-4.06	-2.55	-2.51	-2.80
p 20 → 40	5.90^*	4.38^*	2.06	n/a	n/a	n/a	1.68	1.60	2.16
p 40 → 80	-1.71^*	0.07^*	2.03	1.79	1.78	1.98	2.79	2.74	2.15
p 80 → 160	1.74	1.70	2.01	1.80	1.78	2.48	2.17	2.16	2.06



(a) Out of plane

(b) In-plane

Figure 38. Dynamic nonlinear membrane convergence plots for the final displacements at the origin, including the hybrid method.

Table 13. Dynamic nonlinear membrane convergence data for the mean error magnitudes throughout the simulation, including the hybrid method.

Time Steps	Trapezoid	Hermite	Hybrid
20	0.25	n/a	-0.84
40	-0.35	-1.98	-1.43
80	-0.96	-2.57	-2.04
160	-1.56	-3.17	-2.65
p 20 $\rightarrow$ 40	2.00	n/a	1.97
p 40 $\rightarrow$ 80	2.00	1.98	2.01
p 80 $\rightarrow$ 160	2.00	1.98	2.05

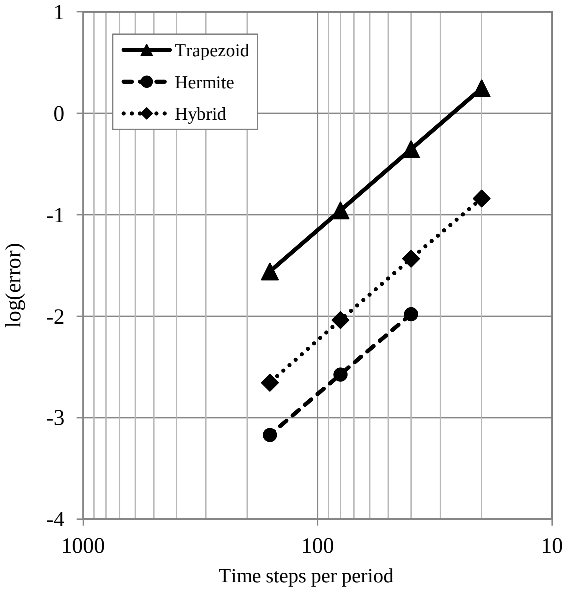


Figure 39. Dynamic nonlinear membrane convergence plot of the mean error magnitudes for all three axes, all spatial nodes, all time steps, including the hybrid method.

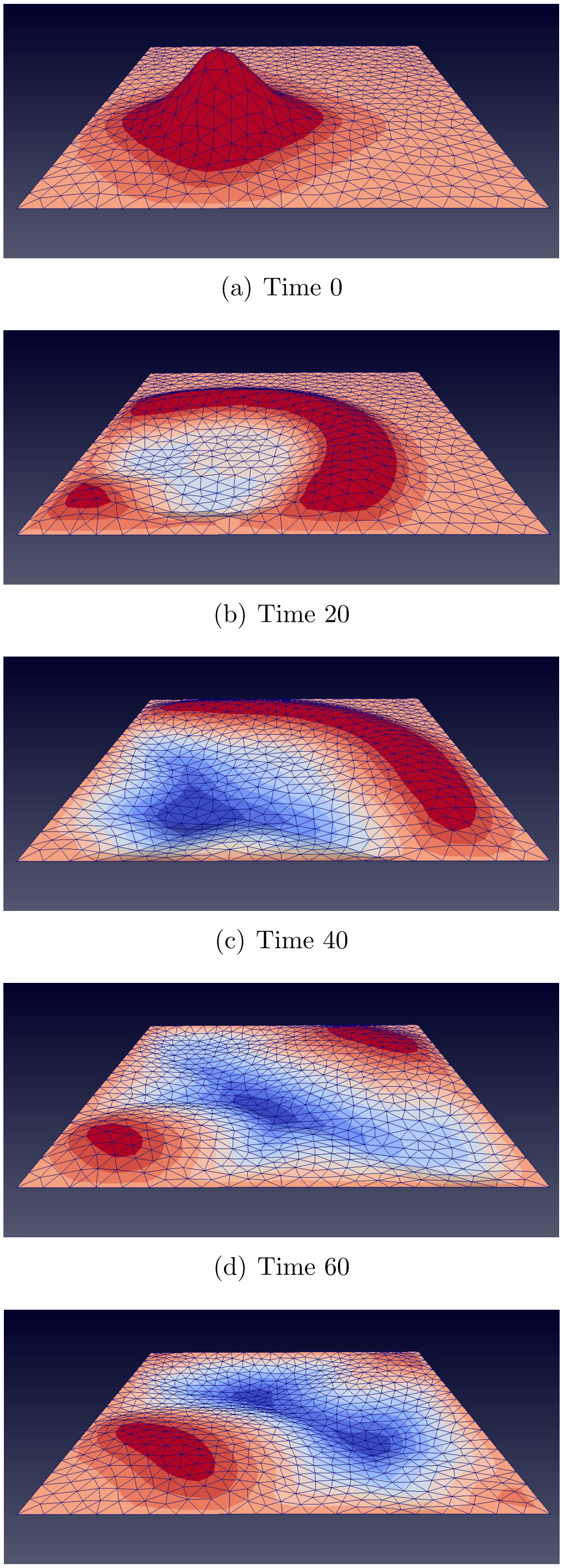
Free Response Analysis.

A case study of the undamped free response of a membrane was performed to challenge the model with a rapid initial transient and wide range of response frequencies. The initial at-rest configuration was a smooth peak slightly off the center of a square membrane (see Eq. 180 and Fig. 40(a)), designed to introduce traveling waves. Initial in-plane displacements were zero.

$$u_3(x_1, x_2) = 0.02\left[\cosh 120(x_1 + 0.015)\cosh 120(x_2 + 0.015)\right]^{-1} \tag{180}$$

The domain's sides were 0.1 meters in length. The unstructured mesh consisted of 700 nodes and 1318 triangular elements, resulting in a representative element size of $h^e = 4.19$ mm (calculated as the length of a side of an equilateral triangle with an area equal to the mesh's mean element area). The same physical constants used in the earlier model verification were retained.

The simulation was performed using the Newmark trapezoid integration method



(a) Time 0

(b) Time 20

(c) Time 40

(d) Time 60

(e) Time 80

Figure 40. Time slices of the free response of a nonlinear membrane.

and the Hermite interpolation method at time steps corresponding to CFL numbers repeatedly halved from 0.8 to 0.025, where the Courant-Friedrich-Levy number $CFL = c\Delta t/h^e$, and the wave speed $c = \sqrt{E/\rho} \approx 45$ m/s. Normalization by use of the CFL number is for intuitive clarity only, since both methods are implicit and therefore technically immune to CFL-driven stability restrictions [79]. The duration of the simulation was 80 time steps with $CFL = 0.8$. For convenience when comparing time series with different time increments, we will refer to these 81 time slices as the "reference times."

The remaining discussion will focus on the out-of-plane response u_3 of the primary mesh node nearest the original peak at $(-0.015, -0.015)$. Since an analytical solution or a proper benchmark solution is not available, a baseline was derived to approximate the true solution. Richardson extrapolation was used to predict the solution at the reference times for each method based on simulations using $CFL = 0.05$ (1,280 time steps) and $CFL = 0.025$ (2,560 time steps). The nearly-coincident extrapolated solutions for the two methods are depicted in Fig. 41. They are essentially converged, as illustrated by plotting the difference in displacement between the two extrapolated solutions in Fig. 42. The largest difference was on the order of 1.0×10^{-6} m, sufficiently small when considering the initial peak's height of 0.02 m. To avoid biasing the results in favor of either technique, the baseline for the following discussion was then calculated as the mean of the two method's extrapolated solutions.

As expected from previous results, the Hermite method's solutions remained closer to the baseline as the time increment was increased. Figure 43 compares the two methods at $CFL = 0.8$. The dashed line of the Hermite method overlies the baseline for much of the simulation. The dispersion of the Newmark method is visible as the solution progressively lags the baseline. A convergence study was performed using the L_2 norms of the errors at the reference times, and the results are plotted in

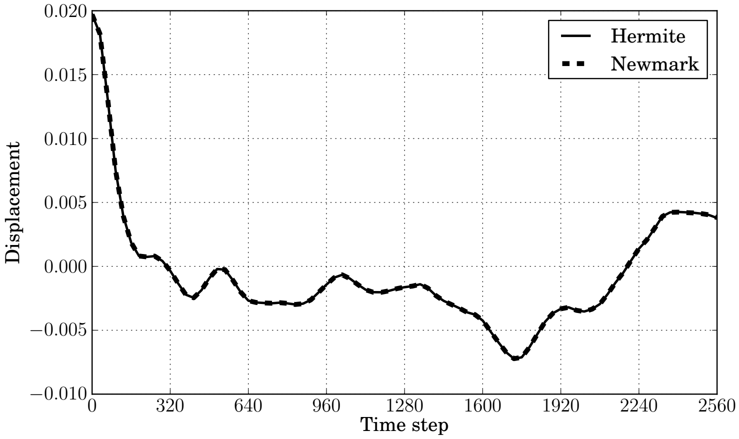


Figure 41. Converged extrapolated solutions for the Hermite and Newmark trapezoidal methods.

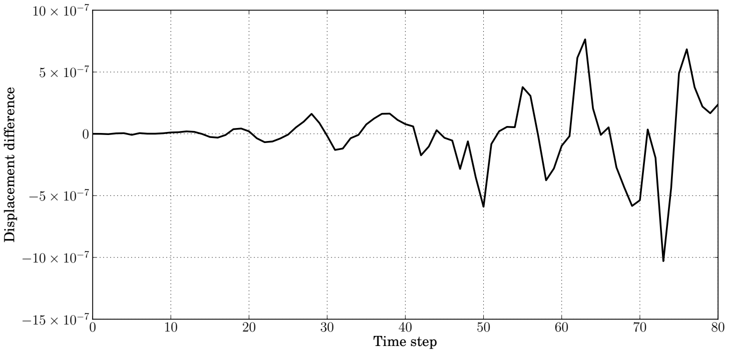


Figure 42. Difference in displacement between the extrapolated solutions of the Hermite and Newmark trapezoidal methods.

Fig. 44. The observed convergence rates of the two methods between the two finest time meshes were 2.06 for the Hermite method and 1.99 for the Newmark method. Thus the results of this case study corroborate those of the verification procedure: the observed rates of convergence of both methods are two, but the Hermite method produces significantly smaller errors.

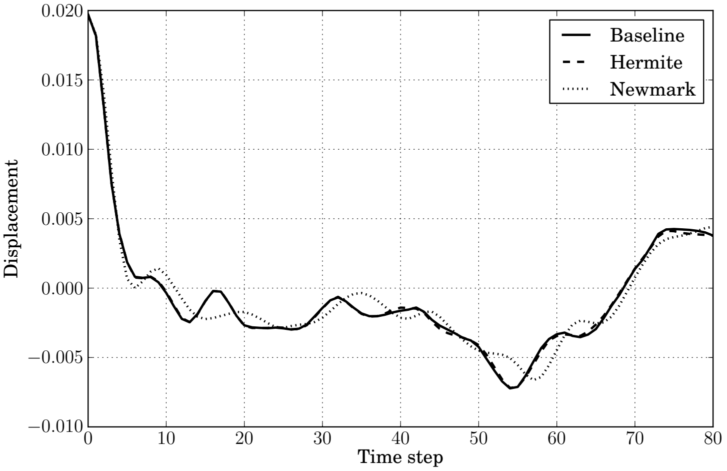


Figure 43. Comparison of the model solutions at $CFL = 0.8$ to the baseline solution.

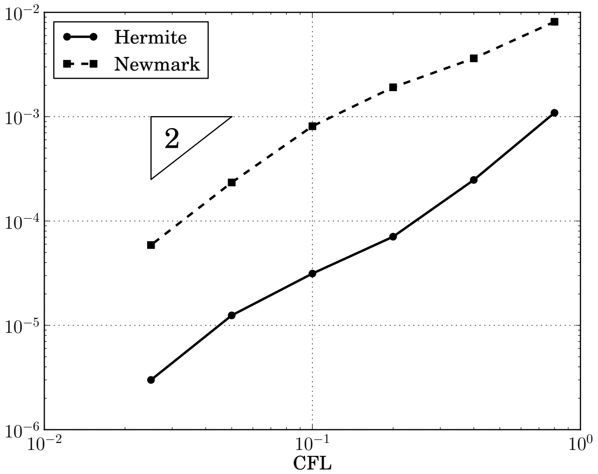


Figure 44. Convergence plots of the dynamic nonlinear membrane.

At the time step sizes presented here, the conditional stability of the Hermite method was not a factor. Performance of both time-marching methods began to suffer significantly at larger time increments – the cost of each time step increased dramatically while accuracy deteriorated. The increased cost was attributed to the decreased accuracy of the initial guesses. The nonlinear solver required more iterations to converge, and convergence became less assured. In our experience the best combination of computational speed and solution accuracy was achieved using the Hermite method with smaller time steps, where the fixed-point iterations converged rapidly. A more rigorous cost comparison would be required to quantify the differences between the two methods and determine optimal time step sizes.

Summary.

This section presented the performance of a dynamic model for geometrically nonlinear membranes. The model incorporated the two previously proposed discretization methods: a jerk constraint-based Hermite time interpolation method, and a staggered-grid point collocation method with grouped nonlinear terms. The Hermite method used the physical meaning of the rate of change of acceleration of the structure to define the jerk constraints. The Newmark trapezoid method was selected for comparison because it is well understood and is the generally accepted standard for structural engineering. The hybrid method was evaluated during verification and found to fill the middle ground between the Newmark and Hermite methods – it enabled larger time step sizes than the pure Hermite method, and the accuracy for out-of-plane displacements was better than the pure Newmark scheme. Two evaluation cases were performed, one forced and one free, by changing time increment sizes for a given mesh. First, the Method of Manufactured Solutions was employed on a hexagonal domain to verify the code and calculate rates of convergence. Second, a free-response case study was evaluated on a square domain. The nearly-converged extrapolated solutions of the two time-marching methods were used to create a baseline solution. The results generally confirmed the behavior of the Hermite time method: minimal frequency distortion and improved accuracy versus the Newmark method. In a notable exception, the observed rate of convergence for the Hermite method fell from four in previous evaluations to two for this membrane model. This rate was observed across both evaluation cases and despite different norms. Overall, for this particular application, the combination of the two proposed discretization schemes was feasible and suitable for obtaining significantly-improved accuracy.

V. Conclusions

The objective of this research was to devise and evaluate novel discretization schemes for the dynamic simulation of membranes. The study was divided into two primary focus areas: (1) linear membranes, and (2) geometrically nonlinear membranes, as defined in Chapter II. Different techniques were developed and tested for the two cases. Throughout this study, the Newmark trapezoid method was selected for comparison because it is well understood and is the generally accepted standard for structural engineering. In this final chapter, the research will be summarized and discussed, contributions will be stated, and avenues for future research will be suggested.

Linear Membrane Dynamics

The Simultaneous Time-Continuous Galerkin (STCG) method was applied to a classical linear membrane model. The method featured trilinear space-time elements and a mixed formulation. The entire space-time domain was discretized, and the simulation's entire solution was obtained by solving the single large, sparse, linear system. Numerical studies indicated observed second-order rates of convergence in both space and time, which is typical of linear elements. Bounded numerical instabilities in the form of high-frequency oscillations were present. However, the lower modes were accurately represented, so a post-solution smoothing procedure was demonstrated that effectively eliminated high-frequency oscillations. Unique to this method, when the time steps exceeded a critical frequency, the solution rapidly collapsed to its resting configuration.

The spurious modes were likely caused by the combination of interpolations used for discretizing the mixed form. As discussed in [146], mixed forms require appropriate matching of interpolations to satisfy stability conditions. Fortunately, for the scheme presented here, the spurious oscillations were bounded and not destructive to the lower-frequency modes. Post-solution filtering removed much of the unwanted content with slight dissipation of the peaks of the lower modes.

Nonlinear Membrane Dynamics

The dynamic nonlinear membrane model was composed of two independently developed schemes: a jerk constraint-based Hermite time interpolation method detailed in Section IV, and a staggered-grid point collocation method with grouped nonlinear terms described in Section IV. Each scheme was rigorously verified and validated through the use of cases with known analytical solutions, manufactured solutions, and experimental data. Finally, the two schemes were combined to form a pure collocation model, which was analyzed in Section IV.

The Hermite method used the physical meaning of the rate of change of acceleration to define the jerk constraints and the resulting unique quintic polynomial interpolation of displacement during a time step. The single-step, implicit scheme was shown to be significantly more accurate than the most common second-order Newmark schemes (central difference and trapezoid). It was proved that the Newmark scheme is a subset of the proposed scheme, and is obtained by appropriately defining the jerk constraints. It was also proved that the scheme is symplectic for a simple harmonic oscillator. Period elongation was derived analytically for a simple harmonic oscillator and shown to be minimal. Through a classical linear analysis, the scheme was found to be conditionally stable, and the limits of stability were analytically derived. The regions of instability were found to be above the Nyquist frequency. The method is not suitable for mathematically stiff systems. However, a novel error estimation scheme was developed which, when used to control time step sizes, enabled the method to successfully solve a stiff system in a numerical demonstration.

The Hermite time interpolation method shows promise for general application to differential equations, as shown by the variety of numerical examples. The method proposed in this paper provides a unique avenue of investigation. Since jerk is not continuous across time steps, the jerk constraints can be modified or optimized separately for each time step or degree of freedom to achieve other desired effects such as algorithmic damping or explicit energy conservation. This feature was illustrated by the hybrid method of Section IV, where the jerk constraints were defined differently in-plane versus out-of-plane.

The proposed spatial discretization scheme accounts for both in-plane and out-of-plane displacements without carrying rotational degrees of freedom. Based on the Method of Weighted Residuals point collocation approach, it employs a staggered grid, grouping of nonlinear terms, and polygon shape functions in a strong-form point collocation formulation. Additional demonstrated capabilities include varying membrane thickness and follower forces, as demonstrated in Section IV. The observed rate of convergence was two for both displacement and strain. Validation against experimental data in the literature showed the method to be accurate until hyperelastic effects begin to appear.

The point collocation scheme is simpler to formulate than conventional nonlinear finite element approaches. Element integration is avoided entirely, a significant benefit since with classical Newton iterations, each integration point (with at least three Gaussian integration points per triangular element) must be computed for each iteration of each time step. The group formulation permits the same treatment in all three axes, again simplifying coding. Overall, the framework of the approach is highly modular and flexible. Any given step can be performed by interchangeable subroutines. The residual subroutine readily accepts different strain-displacement or material models (including nonlinear models).

Lastly, the Hermite time interpolation method and the point collocation spatial scheme were combined to create a novel dynamic membrane model based purely on collocation techniques. Two evaluation cases were considered, one forced and one free. Convergence studies were performed to observe the convergence rates with respect to time step size. First, the Method of Manufactured Solutions was employed on a hexagonal domain to verify the code and calculate rates of convergence. Second, a free-response case study was performed to evaluate the model's ability to capture traveling wave phenomena. The converged extrapolated solutions of the two time-marching methods (Hermite and Newmark trapezoid methods) were averaged to create a baseline solution. Time step sizes were then increased to observe how rapidly the solutions from the two methods deviated from the baseline solution. The results generally confirmed the behavior of the Hermite time method observed in Section IV: minimal frequency distortion and improved accuracy versus the Newmark method. It was notable that the observed rate of convergence for the Hermite method fell from four in the evaluations discussed in Section IV to two for the dynamic membrane simulation discussed in Section IV. Overall, for this particular application, the combination of the two proposed discretization schemes was feasible and suitable for obtaining significantly-improved accuracy.

Contributions

This research effort involved the assessment of a wide variety of numerical methods for use in membrane simulation, and ultimately culminated in a novel, fully three-dimensional, highly modular membrane model that accurately and efficiently simulates the dynamics of geometrically nonlinear membranes. It extends, rigorously investigates, and incorporates specific techniques that are not typically used in structural dynamics. The process of developing and evaluating this model has led to the following specific contributions:

1. Hermite time interpolation method

   - First formulation by strongly enforcing jerk constraints. This approach offers the potential for future optimization by the selection of different constraint formulas.
   - First direct link proven between the jerk constraint-based Hermite interpolation method and the Newmark family of methods.
   - First precise analytical expression for the linear stability limits.
   - First precise analytical expression for dissipation of a simple harmonic oscillator.
   - First local error estimation technique for a Hermite time interpolation. The accurate local error estimation capitalized on the intra-element solution accuracy by using mid-time-step solutions to estimate error.
   - First solution of a stiff system. The local error estimation technique effectively controlled time step size to accurately solve a numerically stiff system.

2. Point collocation model

   - First derivation and application of spatial derivatives of the polygon shape functions proposed in [29].
   - First use of the group finite element method in a point collocation model.
   - First use of a staggered two-dimensional grid approach for nonlinear PDE solution.

3. Geometrically nonlinear membrane model

    - First to incorporate Hermite time interpolation
    - First to incorporate a staggered-grid point collocation spatial discretization

Each of the proposed discretization schemes stands on its own as an effective numerical method for solving nonlinear ordinary or partial differential equations. Therefore, the contributions of this study extend well beyond Micro Air Vehicles, membranes, and structural analysis.

Recommendations for Future Work

Further investigation of the STCG concept should start with determining the precise source of the high-frequency noise and mitigating it intrinsically, so post-solution filtering is not necessary. As discussed in Section III, the results of the static verification indicated a mild instability in the mixed-form solution. Using a displacement formulation rather than a mixed formulation in space could test this source of oscillations, as well as reduce the number of degrees of freedom for the global system. After addressing the high-frequency noise, the temporal stability behavior should be analytically derived. In particular, eigenvalue analysis may explain the tendency of the solution to collapse when the time step size exceeds a certain threshold. Finally, the computational efficiency could be compared to reference schemes such as the linear Newmark and explicit Runge Kutta methods.

Future studies of the Hermite interpolation method can address the following three issues: (1) stability, (2) numerical damping, and (3) computational efficiency. Since only linear stability has been addressed in this research, nonlinear stability behavior could be characterized. The conditional stability of the method is a significant drawback for some applications. Efforts to expand the stability regimes, both intrinsically and using feedback controls, could increase the utility of the method. A component of stability is the numerical damping of high-frequency content. The excellent energy preservation is a useful feature of the method; however, the use of a non-dissipative algorithm for a hyperbolic problem raises the spectre of error propagation and build-up, which can cause a solution to fail. There will likely arise a trade-off between accuracy and damping effectiveness, though by the time the damping is needed, the local solution may have already departed significantly from the exact solution.

For the point collocation spatial discretization scheme of Section IV, further efforts could investigate the sensitivity of the solution to mesh geometry, in particular through analytical *a priori* error estimates and determination of the scheme's order of accuracy. Also, the order of the method could be derived analytically so the *a priori* estimate could be verified by the numerical convergence studies. Free edge boundary conditions for the strong form formulation could potentially be difficult to formulate, so a parallel development of the overall concept (staggered grid with a group FE formulation) from the weak form would probably illuminate a practical path of pursuit. The computational expense of using the proposed scheme with a Jacobian-free nonlinear solver could be compared to a classical linearized finite element formulation using Newton iterations. Finally, application of the method to other systems such as hyperelastic membranes and thin plates offers many interesting opportunities for future work.

Bibliography

[1] Aluru, N. R. "A point collocation method based on reproducing kernel approximations". *International Journal for Numerical Methods in Engineering*, 47(6):1083–1121, 2000.

[2] Arnold, V. I. *Mathematical Methods of Classical Mechanics (Graduate Texts in Mathematics)*. Springer, 2nd edition, 1989.

[3] Ascher, U. M. *Numerical Methods for Evolutionary Differential Equations*. Society for Industrial and Applied Mathematics, 2008.

[4] Atilgan, A. R., D. H. Hodges, M. A. Ozbek, and W. Zhou. "Space-Time Mixed Finite Elements for Rods". *Journal of Sound and Vibration*, 192(3):731–739, 1996.

[5] Atkinson, J. H., J. J. Westerink, and R. A. Luettich. "Two-dimensional dispersion analyses of finite element approximations to the shallow water equations". *International Journal for Numerical Methods in Fluids*, 45(7):715–749, 2004.

[6] Baker, A. H., E. R. Jessup, and T. Manteuffel. "A technique for accelerating the convergence of restarted GMRES". *SIAM Journal on Matrix Analysis and Applications*, 26(4):962–984, 2005.

[7] Baruch, M. and R. Riff. "Hamilton's Principle, Hamilton's Law—6^n correct formulations". *AIAA Journal*, 20(5):687–692, 1982.

[8] Bathe, K. J. *Finite element procedures*. Prentice Hall, 1996.

[9] Bauchau, O. A. and C. H. Hong. "Nonlinear Response and Stability Analysis of Beams Using Finite Elements in Time". *AIAA Journal*, 26(9):1135–1142, 1988.

[10] Belytschko, T., Y. Krongauz, D. Organ, M. Fleming, and P. Krysl. "Meshless methods: an overview and recent developments". *Computer Methods in Applied Mechanics and Engineering*, 139(1-4):3–47, 1996.

[11] Belytschko, T., Y. Y. Lu, and L. Gu. "Element-free Galerkin methods". *International Journal for Numerical Methods in Engineering*, 37(2):229–256, 1994.

[12] Benzi, M. "Preconditioning techniques for large linear systems: a survey". *Journal of Computational Physics*, 182(2):418–477, 2002.

[13] Berman, G. J. and Z. Wang. "Energy-minimizing kinematics in hovering insect flight". *Journal of Fluid Mechanics*, 582:153–168, 2007.

[14] Blazek, J. *Computational Fluid Dynamics: Principles and Applications*. Elsevier, 2nd edition, 2005.

[15] Bonet, J. and R. D. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, 1st edition, 1997.

[16] Borri, M. and C. Bottasso. “A general framework for interpreting time finite element formulations”. *Computational Mechanics*, 13(3):133–142, 1993.

[17] Borri, M., C. Bottasso, and P. Mantegazza. “Basic features of the time finite element approach for dynamics”. *Meccanica*, 27:119–130, 1992.

[18] Borri, M., G. L. Ghiringhelli, M. Lanz, P. Mantegazza, and T. Merlini. “Dynamic Response of Mechanical Systems by a Weak Hamiltonian Formulation”. *Computers and Structures*, 20(1):495–508, 1985.

[19] Braess, D. *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics*. Cambridge University Press, 2nd edition, 2001.

[20] Brandt, A. “Multi-level adaptive solutions to boundary value problems”. *Mathematics of Computation*, 31(138):333–390, 1977.

[21] Briggs, W. L., V. E. Henson, and S. F. McCormick. *A multigrid tutorial*. Society for Industrial and Applied Mathematics, 2nd edition, 2000.

[22] Brown, J. D. “Midpoint rule as a variational-symplectic integrator: Hamiltonian systems”. *Phys. Rev. D*, 73:024001, 2006. URL http://link.aps.org/doi/10.1103/PhysRevD.73.024001.

[23] Burg, C. O. E. and V. K. Murali. “Efficient code verification using the residual formulation of the method of manufactured solutions.” *AIAA Paper 2004-2628. 34th AIAA Fluid Dynamics Conference*. Portland, Oregon, June 2004.

[24] Celik, I. B., U. Ghia, P. J. Roache, C. J. Freitas, H. Coleman, and P. E. Raad. “Procedure for estimation and reporting of uncertainty due to discretization in CFD applications”. *Journal of Fluids Engineering (Special Publication)*, 130, 2008.

[25] Chang-jiang, Liu, Zheng Zhou-lian, He Xiao-ting, Sun Jun-yi, Song Wei-ju, Xu Yun-ping, and Long Jun. “L-P Perturbation Solution of Nonlinear Free Vibration of Prestressed Orthotropic Membrane in Large Amplitude”. *Mathematical Problems in Engineering*, 2010:1–17, 2010.

[26] Christie, I., D. F. Griffiths, A. R. Mitchell, and J. M. Sanz-Serna. “Product Approximation for Non-linear Problems in the Finite Element Method”. *IMA Journal of Numerical Analysis*, 1(3):253–266, 1981.

[27] Combes, S. A. and T. L. Daniel. “Into thin air: Contributions of aerodynamic and inertial-elastic forces to wing bending in the hawkmoth *Maduca sexta*”. *Journal of Experimental Biology*, 206(17):2999–3006, 2003.

[28] Cook, R. D., D. S. Malkus, M. E. Plesha, and R. J. Witt. *Concepts and Applications of Finite Element Analysis.* John Wiley & Sons, Inc., 4th edition, 2002.

[29] Dasgupta, G. "Interpolants within convex polygons: Wachspress' shape functions". *Journal of Aerospace Engineering*, 16(1):1–8, 2003.

[30] Dash, R. R., Mohanty S. C., and T. Rout. "Evaluation of Non Linear Response of Beams by Symplectic Integration Method". *International Journal of Engineering Studies*, 2(2):119–134, 2010.

[31] Delingette, H. "Triangular springs for modeling nonlinear membranes". *IEEE Transactions on Visualization and Computer Graphics*, 14(2):329–341, 2008.

[32] Doma, S. B., I. H. El-Sirafy, M. M. El-Borai, and A. H. El-Sharif. "Perturbation Treatment for the Vibrations of a Circular Membrane Subject to a Restorative Force". *Alexandria Journal of Mathematics*, 1(1):40–52, 2010.

[33] Doyle, J. F. *Wave Propagation in Structures: Spectral Analysis Using Fast Discrete Fourier Transforms.* Springer, 2nd edition, 1997.

[34] Ferguson, L., E. Aulisa, P. Seshaiyer, and R. Gordnier. "Computational Modeling of Highly Flexible Membrane Wings in Micro Air Vehicles". *Proceedings of the 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, AIAA 2006-1661. Newport, Rhode Island, 2006.

[35] Fichter, W. B. *Some Solutions for the Large Deflections of Uniformly Loaded Circular Membranes.* NASA Technical Paper 3658, Langley Research Center, Hampton, VA, 1997.

[36] Finn, D. L. *MA 323 Geometric Modeling, Course Notes: Day 09, Quintic Hermite Interpolation, Dec 13, 2004.* Department of Mathematics, Rose-Hulman Institute of Technology. URL http://www.rose-hulman.edu/ finn/C-CLI/Notes/day09.pdf, 2004.

[37] Fletcher, C. A. J. "The group finite element formulation". *Computer Methods in Applied Mechanics and Engineering*, 37(2):225–244, 1983.

[38] Fletcher, C. A. J. and K. Srinivas. "Time-split finite element methods in physical and generalized coordinates". R. H. Gallagher, G. Carey, J. T. Oden, and O. C. Zienkiewicz (editors), *Finite Elements in Fluids*, volume 6: Finite Elements and Flow Problems of *Wiley Series in Numerical Methods in Engineering*, chapter 4, 115–133. John Wiley & Sons, 1985.

[39] Foreman, M. G. G. "An analysis of the "wave equation" model for finite element tidal computations". *Journal of Computational Physics*, 52(2):290–312, 1983.

[40] Gee, M. W., C. T. Kelley, and R. B. Lehoucq. "Pseudo-transient continuation for nonlinear transient elasticity". *International Journal for Numerical Methods in Engineering*, 78(10):1209–1219, 2009.

[41] Gee, M. W. and R. S. Tuminaro. *Nonlinear algebraic multigrid for constrained solid mechanics problems using Trilinos.* Tech. Rep. SAND2006-2256, Sandia National Laboratories, 2006.

[42] Gelfand, I. M, S. V Fomin, and R. A Silverman. *Calculus of variations.* Prentice-Hall Englewood Cliffs, NJ, 1963.

[43] Geuzaine, C. and J.-F. Remacle. "Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities". *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.

[44] Goldstein, H. *Classical mechanics.* Addison-Wesley Pub. Co., 1980.

[45] Gordnier, R. E. "High fidelity computational simulation of a membrane wing airfoil". *Journal of Fluids and Structures*, 25(5):897–917, 2009.

[46] Graff, K. F. *Wave motion in elastic solids.* Dover Publications, 1991.

[47] Griewank, A. and A. Walther. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation.* Society for Industrial and Applied Mathematics, 2nd edition, 2008.

[48] Hairer, E., C. Lubich, and G. Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations.* Springer, 2006.

[49] Hairer, E. and G. Wanner. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems.* Springer, 2nd edition, 1996.

[50] Hamming, R. W. *Numerical Methods for Scientists and Engineers.* Dover Publications, 2nd edition, 1987.

[51] Hencky, H. "On the stress state in circular plates with vanishing bending stiffness". *Zeitschrift für Mathematik und Physik*, 63(3):311–317, 1915.

[52] Henson, V. E. "Multigrid methods nonlinear problems: an overview". *Proc. SPIE 5016, 36.* 2003.

[53] Hesthaven, J. S. and T. Warburton. *Nodal Discontinuous Galerkin Methods.* Springer Science+Business Media, LLC, 2008.

[54] Hilber, H. M., T. J. R. Hughes, and R. L. Taylor. "Improved numerical dissipation for time integration algorithms in structural dynamics". *Earthquake Engineering & Structural Dynamics*, 5(3):283–292, 1977.

[55] Hirsch, C. *Numerical Computation of Internal & External Flows.* John Wiley & Sons, Ltd., 2007.

[56] Hodges, D. H. and R. R. Bless. "Weak Hamiltonian Finite Element Method for Optimal Control Problems". *Journal of Guidance and Control*, 14(1):148–156, 1991.

[57] Hodges, D. H., W. Yu, and M. J. Patil. "Geometrically-Exact, Intrinsic Theory for Dynamics of Moving Composite Plates". *International Journal of Solids and Structures*, 46:2036–2042, 2009.

[58] Hughes, T. J. R. *The Finite Element Method.* Prentice-Hall, Inc., 1987.

[59] Hulbert, G. M. "Time finite element methods for structural dynamics". *International Journal for Numerical Methods in Engineering*, 33(2):307–331, 1992.

[60] Hulbert, G. M. and T. J. R. Hughes. "Space-time finite element methods for second-order hyperbolic equations". *Computer Methods in Applied Mechanics and Engineering*, 84(3):327–348, 1990.

[61] Idelsohn, S. R., E. Oñate, N. Calvo, and F. Del Pin. "The meshless finite element method". *International Journal for Numerical Methods in Engineering*, 58(6):893–912, 2003.

[62] Idesman, A. V. "A new high-order accurate continuous Galerkin method for linear elastodynamics problems". *Computational Mechanics*, 40(2):261–279, 2007.

[63] Jenkins, C. H. "Nonlinear dynamic response of membranes: State of the art–Update". *Applied Mechanics Reviews*, 49(10S):S41–S48, 1996.

[64] Jenkins, C. H. and J. W. Leonard. "Nonlinear dynamic response of membranes: State of the art". *Applied Mechanics Reviews*, 44(7):319–328, 1991.

[65] Jones, E., T. Oliphant, P. Peterson, et al. "SciPy: Open source scientific tools for Python", 2001–. URL http://www.scipy.org/.

[66] Jones, J. E. and C. S. Woodward. "Newton-Krylov-multigrid solvers for large-scale, highly heterogeneous, variably saturated flow problems". *Advances in Water Resources*, 24(7):763–774, 2001.

[67] Kaliakin, V. N. *Introduction to Approximate Solution Techniques, Numerical Modeling, and Finite Element Methods.* CRC Press, 2002.

[68] Kane, C., J. E. Marsden, M. Ortiz, and M. West. "Variational Integrators and the Newmark Algorithm for Conservative and Dissipative Mechanical Systems". *Internat. J. Numer. Methods Engrg*, 49:1295–1325, 1999.

[69] Kelley, C. T. *Iterative methods for linear and nonlinear equations.* Society for Industrial and Applied Mathematics, 1995.

[70] Kelley, C. T. and D. E. Keyes. "Convergence analysis of pseudo-transient continuation". *SIAM Journal of Numerical Analysis*, 35(2):508–523, 1998.

[71] Ketter, R. and S. Prawel. *Modern Methods of Engineering Computation.* McGraw-Hill, Inc., 1969.

[72] Kharevych, L., W. Yang, Y. Tong, E. Kanso, J. E. Marsden, P. Schröder, and M. Desbrun. "Geometric, variational integrators for computer animation". *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, SCA '06, 43–51. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2006.

[73] Kilian, A. and J. Ochsendorf. "Particle-spring systems for structural form finding". *IASS Journal of the International Association for Shell and Spatial Structures*, 46(2):77–85, 2005.

[74] Knabner, P. and L. Angermann. *Numerical Methods for Elliptic and Parabolic Partial Differential Equations.* Springer, 2nd edition, 2003.

[75] Knoll, D. A. and D. E. Keyes. "Jacobian-free Newton-Krylov methods: A survey of approaches and applications". *Journal of Computational Physics*, 193(2):357–397, 2004.

[76] Krysl, P. and T. Belytschko. "Analysis of thin shells by the element-free Galerkin method". *International Journal of Solids and Structures*, 33(20-22):3057–3080, 1996.

[77] Lambert, J. D. *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem.* Wiley, 1st edition, 1991.

[78] Lanczos, C. *The variational principles of mechanics.* Dover Publications, 1970.

[79] Laney, C. B. *Computational Gasdynamics.* Cambridge University Press, 1998.

[80] Leigh, E. J. and D. L. Kunz. "Simulation of a Moving Elastic Beam Using Hamiltons Weak Principle". *AIAA Journal*, 45(2):471–476, 2007.

[81] Leok, M. and T. Shingel. "Prolongation-Collocation Variational Integrators". *arXiv:1101.1995v1*, 2011.

[82] Leung, A. Y. T. and S. G. Mao. "Symplectic integration of an accurate beam finite element in nonlinear vibration". *Computers & Structures*, 54(6):1135–1147, 1995.

[83] Leung, A. Y. T., B. Zhu, J. Zheng, and H. Yang. "A trapezoidal Fourier p-element for membrane vibrations". *Thin-Walled Structures*, 41(5):479–491, 2003.

[84] Lew, A., J. E. Marsden, M. Ortiz, and M. West. "Variational time integrators". *International Journal for Numerical Methods in Engineering*, 60(1):153–212, 2004.

[85] Li, B. Q. *Discontinuous Finite Elements in Fluid Dynamics and Heat Transfer.* Springer, 2006.

[86] Lian, Y., W. Shyy, P. G. Ifju, and E. Verron. "Membrane Wing Model for Micro Air Vehicles". *AIAA Journal*, 41(12):2492–2494, 2003.

[87] Lian, Y., W. Shyy, D. Viieru, and B. Zhang. "Membrane wing aerodynamics for micro air vehicles". *Progress in Aerospace Sciences*, 39(6-7):425–465, 2003.

[88] Liu, S., J. B. Haddow, and S. Dost. "A variational approach to a circular hyperelastic membrane problem". *Acta mechanica*, 99(1):191–200, 1993.

[89] Mackey, D. S. and N. Mackey. *On the determinant of symplectic matrices.* Numerical Analysis Report 422, Manchester Centre for Computational Mathematics, Manchester, England, February 2003.

[90] Malik, A. M. A and B. M.S.A Qureshi. "Vibration Analysis of Flapping Wing Micro Air Vehicle Using Finite Element Methods". *Proceedings of the World Congress on Engineering, WCE 2010*, volume 2. 2010.

[91] Meirovitch, L. *Analytical Methods in Vibrations.* The Macmillan Company, 1967.

[92] Mulder, W. A. "Spurious modes in finite-element discretizations of the wave equation may not be all that bad". *Applied Numerical Mathematics*, 30:425–445, 1999. ACM ID: 329581.

[93] Ou, K., P. Castonguay, and A. Jameson. "3D Flapping Wing Simulation with High-Order Spectral Difference Method on Deformable Mesh". *AIAA Paper 2011-1316, 49th AIAA Aerospace Science Meeting, Orlando, FL.* January 2011.

[94] Owren, B. and H. H. Simonsen. "Alternative integration methods for problems in structural dynamics". *Computer Methods in Applied Mechanics and Engineering*, 122(1-2):1–10, 1995.

[95] Pai, P. F. *Highly Flexible Structures: Modeling, Computation, and Experimentation.* American Institute of Aeronautics and Astronautics, 2007.

[96] Pai, P. F. and L. G. Young. "Fully nonlinear modeling and analysis of precision membranes". *International Journal of Computational Engineering Science*, 4(1):19–65, 2003.

[97] Pauletti, R. M. O., D. M. Guirardi, and T. E. C. Deifeld. "Argyris' Natural Membrane Finite Element Revisited". E. Oñate and B. Kröpling (editors), *Textile Composites and Inflatable Structures*, 335–344. CIMNE, Barcelona, 2005.

[98] Payen, D. J. and K. J. Bathe. "The use of nodal point forces to improve element stresses". *Computers & Structures*, 89(5-6):485–495, 2011.

[99] Persson, P.-O. and G. Strang. "Smoothing by Savitzky-Golay and Legendre filters". Joachim Rosenthal and David S. Gilliam (editors), *IMA Volume on Math. Systems Theory in Biology, Comm., Comp., and Finance*, volume 134, 301–316. Springer, 2003.

[100] Peters, D. A. and A. P. Izadpanah. "hp-version finite elements for the space-time domain". *Computational Mechanics*, 3(2):73–88, 1988.

[101] Peterson, S., F. Charbel, and R. Tezaur. "A space-time discontinuous Galerkin method for the solution of the wave equation in the time-domain". *International Journal for Numerical Methods in Engineering*, 78(3):275–295, 2009.

[102] Petyt, M. *Introduction to Finite Element Vibration Analysis.* Cambridge University Press, 1990.

[103] Pozrikidis, C. *Numerical Computation in Science and Engineering.* Oxford University Press, 1998.

[104] Press, W. H., S. A. Teukolsky, and B. P. Vetterling, W. T.and Flannery. *Numerical Recipes: The Art of Scientific Computing.* Cambridge University Press, 3rd edition, 2007.

[105] Proakis, J. G. and D. G. Manolakis. *Digital signal processing: principles, algorithms, and applications.* Macmillan, 1992.

[106] Pujara, P. and T. J. Lardner. "Deformations of elastic membranes–Effect of different constitutive relations". *Zeitschrift für Angewandte Mathematik und Physik (ZAMP)*, 29(2):315–327, 1978.

[107] Rao, S. S. *The Finite Element Method in Engineering.* Butterworth-Heinemann, 3rd edition, 1999.

[108] Rao, S. S. *Vibrations of Continuous Systems.* John Wiley & Sons, Inc., 2007.

[109] Razavi, S. H., A. Abolmaali, and M. Ghassemieh. "A weighted residual parabolic acceleration time integration method for problems in structural dynamics". *Computational Methods in Applied Mathematics*, 7(3):227–238, 2007.

[110] Reddy, J. N. *An Introduction to the Finite Element Method.* McGraw-Hill, Inc., 1984.

[111] Reddy, J. N. *Theory and Analysis of Elastic Plates.* Taylor & Francis, 1st edition, 1999.

[112] Roache, P. J. *Verification and Validation in Computational Science and Engineering.* Hermosa Publishers, 1998.

[113] Rojratsirikul, P., Z. Wang, and I. Gursul. "Unsteady fluid-structure interactions of membrane airfoils at low Reynolds numbers". *Experiments in Fluids*, 46(5):859–872, 2009.

[114] Ruffin, C. and R. L. King. "The Analysis of Hyperspectral Data Using Savitzky-Golay Filtering-Theoretical Basis (Part 1)". *Geoscience and Remote Sensing Symposium, 1999 IEEE International, IGARSS 1999*, 2:756–758, 1999.

[115] Salari, K. and P. Knupp. *Code verification by the method of manufactured solutions.* Tech. Rep. SAND2000-1444, Sandia National Laboratories, 2000.

[116] Sanz-Serna, J. M. and M. P. Calvo. *Numerical Hamiltonian Problems.* Chapman and Hall/CRC, 1st edition, 1994.

[117] Schilling, R. J. and S. L. Harris. *Applied Numerical Methods for Engineers using MATLAB and C.* Brooks/Cole, 2000.

[118] Shabana, A. A. *Vibration of Discrete and Continuous Systems.* Springer, 2nd edition, 1997.

[119] Shyy, W., P. Ifju, and D. Viieru. "Membrane Wing-Based Micro Air Vehicles". *Applied Mechanics Reviews*, 58(4):283–301, 2005.

[120] Silva, C. W. de. *Computer Techniques in Vibration.* CRC Press, 1st edition, 2007.

[121] Sims, T. W., A. N. Palazotto, and A. Norris. "A Structural Dynamic Analysis of a Manduca Sexta Forewing". *International Journal of Micro Air Vehicles*, 2(3):119–140, 2010.

[122] Small, M. K. and W. D. Nix. "Analysis of the accuracy of the bulge test in determining the mechanical properties of thin films". *Journal of Materials Research*, 7(6):1553–1563, 1992.

[123] Sprott, J. C. "Some simple chaotic jerk functions". *American Journal of Physics*, 65(6):537–543, 1997.

[124] Stanford, B. *Aeroelastic analysis and optimization of membrane micro air vehicle wings.* Ph.D. thesis, University of Florida, 2008.

[125] Stanford, B. and P. Ifju. "The Validity Range of Low Fidelity Structural Membrane Models". *Experimental Mechanics*, 48(6):697–711, 2008.

[126] Stanford, B., P. Ifju, R. Albertani, and W. Shyy. "Fixed membrane wings for micro air vehicles: Experimental characterization, numerical modeling, and tailoring". *Progress in Aerospace Sciences*, 44(4):258–294, 2008.

[127] Strikwerda, J. C. and J. M. Considine. *Deformation of a Membrane Under Uniform Static Pressure.* Computer sciences technical report 1177, University of Wisconsin-Madison, Computer Sciences Department, 1993.

[128] Swartz, S. M., J. Iriarte-Diaz, D. K. Riskin, A. Song, X. Tian, D. J. Willis, and K. S. Breuer. "Wing structure and the aerodynamic basis of flight in bats". *American Institute of Aeronautics and Astronautics Journal*, 42:1–10, 2007.

[129] SymPy Development Team. *SymPy: Python library for symbolic mathematics*, 2009. URL http://www.sympy.org.

[130] Thompson, L. L. and P. M. Pinsky. "Complex Wavenumber Fourier Analysis of the P-Version Finite Element Method". *Computational Mechanics*, 13(4):255–275, 1994.

[131] Van der Vorst, H. A. *Iterative Krylov Methods for Large Linear Systems.* Cambridge University Press, 1st edition, 2003.

[132] Van Gelder, A. "Approximate simulation of elastic membranes by triangulated spring meshes". *Journal of Graphics Tools*, 3(2):21–42, 1998.

[133] Virgin, L. N. *Vibration of axially loaded structures.* Cambridge University Press, 2007.

[134] Volino, P., N. Magnenat-Thalmann, and F. Faure. "A simple approach to nonlinear tensile stiffness for accurate cloth simulation". *ACM Trans. Graph.*, 28(4):105:1–105:16, 2009.

[135] Walker, H. F. and P. Ni. "Anderson acceleration for fixed-point iterations". *SIAM Journal on Numerical Analysis*, 49(4):1715–1735, 2011.

[136] Weinstock, R. *Calculus of variations: with applications to physics and engineering.* Dover Publications, 1974.

[137] West, M. *Variational integrators.* Dissertation, California Institute of Technology, 2004.

[138] Willmott, A. and C. Ellington. "The mechanics of flight in the hawkmoth Manduca Sexta". *Journal of Experimental Biology*, 200:2705–2722, 1997.

[139] Wootton, R. J., R. C. Herbert, P. G. Young, and K. E. Evans. "Approaches to the structural modelling of insect wings". *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 358(1437):1577–1587, 2003.

[140] Wriggers, P. *Nonlinear Finite Element Methods.* Springer, 2008.

[141] Wu, B., X. Du, and H. Tan. "A three-dimensional FE nonlinear analysis of membranes". *Computers & Structures*, 59(4):601–605, 1996.

[142] Xie, W.-C. *Differential Equations for Engineers.* Cambridge University Press, 2010.

[143] Yin, B. and H. Luo. "Effect of wing inertia on hovering performance of flexible flapping wings". *Physics of Fluids*, 22(11):111902–1–10, 2010.

[144] Yoshida, H. "Construction of higher order symplectic integrators". *Physics Letters A*, 150(5-7):262–268, 1990.

[145] Zienkiewicz, O. C., R. L. Taylor, S. J. Sherwin, and J. Peiro. "On discontinuous Galerkin methods". *International Journal for Numerical Methods in Engineering*, 58(8):1119–1148, 2003.

[146] Zienkiewicz, O. C., R. L. Taylor, and J. Z. Zhu. *The Finite Element Method: Its Basis and Fundamentals.* Butterworth-Heinemann, 6th edition, 2005.

Vita

Lieutenant Colonel Kyle F. Kolsti was born and raised in Austin, Texas. He earned a B.S. in Civil Engineering at the University of Texas at Austin in May 1993, graduating with Highest Honors, followed by an M.S. in Civil Engineering at the University of Washington in August 1995. His engineering career was temporarily paused after he obtained a private pilot's license and decided to join the United States Air Force. He was commissioned a Second Lieutenant after graduation from Officer Training School in October 1996.

Lieutenant Colonel Kolsti was the Distinguished Graduate from Undergraduate Pilot Training Class 98-05, Columbus AFB, MS. He elected to fly F-16s and flew operational tours at Shaw AFB, SC and Kunsan AB, Republic of Korea. He then returned to Texas to be a T-38 instructor pilot for pilot trainees at Laughlin AFB.

Still intrigued by the technical aspects of flight, Lieutenant Colonel Kolsti attended the USAF Test Pilot School at Edwards AFB, CA, and graduated with Class 05B in June 2006. During the following three-year tour at Eglin AFB, FL, he flew a wide variety of flight test missions and deployed to Baghdad, Iraq. In August 2009 he entered the Graduate School of Engineering and Management at the Air Force Institute of Technology. Upon graduation he will be assigned to the Air Force Research Laboratories (AFRL) at Wright-Patterson AFB, Ohio.

Lieutenant Colonel Kolsti is a senior pilot with over 2000 hours in more than 30 different types of aircraft.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
13-09-2012

2. REPORT TYPE
PhD Dissertation

3. DATES COVERED *(From - To)*
August 2009 to September 2012

4. TITLE AND SUBTITLE
Novel Discretization Schemes for the Numerical Simulation of Membrane Dynamics

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Kyle F. Kolsti, Lt Col, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/ENY)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/DS/ENY/12-03

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
POC: Dr. David Stargel,
Phone: (703) 696-6961 Email: david.stargel@afosr.af.mil
Air Force Office of Scientific Research
875 N. Randolph, Suite 325, Rm 3112
Arlington, VA 22205

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Motivated by the demands of simulating flapping wings of Micro Air Vehicles, novel numerical methods were developed and evaluated for the dynamic simulation of membranes. For linear membranes, a mixed-form time-continuous Galerkin method was employed using trilinear space-time elements, and the entire space-time domain was discretized and solved simultaneously. For geometrically nonlinear membranes, the model incorporated two new schemes that were independently developed and evaluated. Time marching was performed using quintic Hermite polynomials uniquely determined by end-point jerk constraints. The single-step, implicit scheme was significantly more accurate than the most common Newmark schemes. For a simple harmonic oscillator, the scheme was found to be symplectic, frequency-preserving, and conditionally stable. Time step size was limited by accuracy requirements rather than stability. The spatial discretization scheme employed a staggered grid, grouping of nonlinear terms, and polygon shape functions in a strong-form point collocation formulation. Validation against existing experimental data showed the method to be accurate until hyperelastic effects dominate.

15. SUBJECT TERMS
membrane, nonlinear, numerical methods, point collocation

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES
U	U	U	UU	159

19a. NAME OF RESPONSIBLE PERSON
Donald L. Kunz, Ph.D., P.E.

19b. TELEPHONE NUMBER *(Include area code)*
(937) 255-3636, ext 4548 (Donald.Kunz@afit.edu)